Yiteng (Arden) Huang
Jacob Benesty
(Eds.)

Audio Signal Processing

For Next-Generation Multimedia Communication Systems

Kluwer Academic Publishers

AUDIO SIGNAL PROCESSING FOR NEXT-GENERATION MULTIMEDIA COMMUNICATION SYSTEMS

This page intentionally left blank

AUDIO SIGNAL PROCESSING FOR NEXT-GENERATION MULTIMEDIA COMMUNICATION SYSTEMS

Edited by
YITENG (ARDEN) HUANG
Bell Laboratories, Lucent Technologies

JACOB BENESTY
Université du Québec, INRS-EMT

KLUWER ACADEMIC PUBLISHERS
NEW YORK, BOSTON, DORDRECHT, LONDON, MOSCOW

eBook ISBN: 1-4020-7769-6
Print ISBN: 1-4020-7768-8

Created in the United States of America

Contents

3

4

Part II Acoustic Echo Cancellation

5

Preface

This book has its origins in the Multimedia Communications Research Center at Bell Laboratories. It will likely be the last thing coming from this Center, since it no longer exists. This book reflects our vision on next-generation multimedia communication systems, shared by many researchers who worked with us at a time in the last several years in the Bell Labs' Acoustics and Speech Research Department. Before falling apart in the recent relentless telecommunication firestorm, this department had a legendary history of producing ground-breaking technological advances that have contributed greatly to the success of fundamental voice and advanced multimedia communications technologies. Its scientific discoveries and innovative inventions made a remarkable impact on the cultures within both the industrial and academic communities. But due to the declining financial support from its parent company, Lucent Technologies, Bell Labs decided to lower, if not completely stop, the effort on developing terminal technologies for multimedia communications. As a result, many budding and well established acoustics and speech scientists left Bell Labs but soon resumed their painstaking and rewarding research elsewhere. Their spirit for innovation and passion for what they believe did not disappear; quite to the contrary it will likely live on (this book is the best evidence). The "flame" of their curiosity and enthusiasm will keep burning and will be transmitted to the next generation without a doubt. The idea of editing such a book was triggered early in 2003 when we discussed how to continue our ongoing collaboration after we could no longer work together in the same place. By inviting our former colleagues and collaborators to share their thoughts and state-of-the-art research results, we hope to bring the readers to the very frontiers of audio signal processing for next-generation multimedia communication systems.

We deeply appreciate the efforts, interest, and enthusiasm of all of the contributing authors. Thanks to their cooperation, editing this book turned out to be a very pleasant experience for us. We are grateful to our former department head, Dr. Biing-Hwang (Fred) Juang (currently Motorola Foundation Chair Professor at the Georgia Institute of Technology) for his role in encouraging our research through the department's most difficult time. Without his inspiration, this book would have never been published. We would also like to thank our former colleagues, Dr. Mohan Sondhi, Dr. James West, Dr. Steven Gay, Dr. Dennis Morgan, Dr. Frank Soong, and many others, for their friendship and support over the years. For the preparation of the book, we would like to thank Alex Greene, our production editor, and Melissa Sullivan at Kluwer Academic Publishers.

YITENG (ARDEN) HUANG, JACOB BENESTY

Contributing Authors

Robert Aichner
University of Erlangen–Nuremberg

Carlos Avendano
Creative Advanced Technology Center

Jacob Benesty
Université du Québec, INRS-EMT

Herbert Buchner
University of Erlangen–Nuremberg

Jingdong Chen
Bell Labs, Lucent Technologies

Eric J. Diethorn
Avaya Labs

Gary W. Elko
Avaya Labs

Volker Fischer
Darmstadt University of Technology

Tomas Gänsler
Agere Systems

Yiteng (Arden) Huang
Bell Labs, Lucent Technologies

Walter Kellermann
University of Erlangen–Nuremberg

Achim Kuntz
University of Erlangen–Nuremberg

Jens Meyer
MH Acoustics

Rudolf Rabenstein
University of Erlangen–Nuremberg

Markus Rupp
Vienna University of Technology

Gerald Schuler
Fraunhofer AEMT

Sascha Spors
University of Erlangen–Nuremberg

Heinz Teutsch
University of Erlangen–Nuremberg

Chapter 1

INTRODUCTION

Yiteng (Arden) Huang
Bell Laboratories, Lucent Technologies
arden @ research.bell-labs.com

Jacob Benesty
Université du Québec, INRS-EMT
benesty @ inrs-emt.uquebec.ca

1. MULTIMEDIA COMMUNICATIONS

Modern communications technology has become a part of our daily experience and has dramatically changed the way we live, receive education, work, and relate to each other. Nowadays, communications are already fundamental to smooth functioning of the contemporary society and our individual lives. The expeditious growth in our ability to communicate was one of the most revolutionary advancements in our society over the last century, particularly the last two decades.

In recent progress of the communication revolution, we observe four technical developments that have altered the entire landscape of the telecommunication marketplace. The first is the proliferation of data transfer rate. Thanks to technological breakthroughs in fiber optics, the data transfer rate was boosted from around 100 Mbps (Megabits per second) for one single fiber at the beginning of the 1980s to today's 400 Gbps (Gigabits per second). The capacity of optical fiber has increased by a factor of four thousand in just twenty years, exceeding Moore's law (which describes the growth in the number of transistors per square inch on integrated circuits.) The second is the ubiquity of packet switched networks driven by the ever-growing popularity of the Internet and the World-Wide Web. The invention of the Internet and the World-Wide Web created a common platform for us to share highly diverse information in a relatively unified manner. Ubiquitous packet switched networks make the world more intertwined than ever from an economical, cultural, and even political perspective. Compared to conventional circuit switched networks, packet switched networks are more cost effective and more efficient. Furthermore, adding new services and applications is obviously easier and more flexible on packet switched networks than on circuit switched networks. As a result, the convergence of circuit and packet switched networks has emerged and the trend will continue into the future. The third is the wide deployment of wireless communications. A decade ago most people barely knew of wireless personal communications. In those days, it was basically a techie's vision. But today, it is no longer an electrical engineer's dream. Wireless communications has been well accepted by the public and its business is growing bigger every day and everywhere. The wireless communications technology has evolved from first-generation analog systems to second-generation digital systems, and will continue advancing into its third generation, which is optimized for both voice and data communication services. The last but not least significant development is the escalating demand for broadband access through such connections as digital subscriber line (DSL) or cable to the Internet. Broadband access enables a large number of prospective bandwidth-consuming services that will potentially make our work more productive and our leisure more rewarding. These developments are shaking the foundations of the telecommunications industry and it can be foreseen that tomorrow's communications will be carried out over fast, high-capacity packet-switched networks with mobile, broadband access anytime and anywhere.

Packet switched networks so far have achieved great success for transferring data in an efficient and economical manner. Data communications have enabled us to acquire timely information from virtually every corner of the world. Our intuition may tell that the faster a network could be, the more favorable it becomes. But there is a lack of perceived benefit from paying more to gain another further quantum leap to even faster networks. We believe that it is more imperative and more urgent to introduce innovative communication services that keep up with the aforementioned four developments in communication technologies. Multimedia communications for telecollaboration (for example teleconferencing, distant learning, and telemedicine) over packet switched networks is one of the most promising choices. The features that it introduces will more profoundly enhance peoples' life in the way they communicate, and will bring remarkable values to service providers, enterprises, and end-users.

For a collaboration, full-scale interaction and a sense of immersion are essential to put the users in control and to attain high collaborative productivity in spite of long distances. In this case, not only messages would be exchanged, but also experiences (sensory information) need to be shared. Experiences are inherently composed of a number of different media and advanced multimedia technologies are crucial to the successful implementation of a telecollaboration system. The desire to share experiences has been and will continue to be the motivating factor in the development of exciting multimedia technologies.

Multimedia communication differs from traditional communication modes in that it is no longer constrained by one given medium. It selects appropriate media according to the content and combines messages and experiences together. With enriched experiences, remote environments can be reproduced as faithfully as possible so that local users can make full use of both their binaural hearing and binocular vision. Such an immersive interface makes it easier to determine who is talking and helps understand better what is being discussed, particularly when there are multiple participants. Full-scale interaction differentiates collaboration from exhibition although both are possibly powered by multimedia. Interaction establishes two channels of information flow from and to a user, which makes communication more effective. This can be well recognized by considering the effectiveness of a lecture with and without allowing the audience to raise questions. Full-scale interaction and a sense of immersion are indeed the two most important features of collaboration, and we cannot afford to intentionally sacrifice them anymore in building next-generation communication systems.

Evidently, the most powerful way to conduct full-scale interaction and to create a sense of immersion in telecollaboration is with both visual and audio properly involved. But due to space limitation and the authors' expertise, this book will focus exclusively on the processing, transmission, and presentation of audio and acoustic signals in multimedia communications for telecollaboration. The ideal acoustic environment that we are pursuing is referred to as immersive acoustics, which demands at least full-duplex, hands-free, and spatial perceptibility. As a result, we confront remarkable challenges to address a number of complicated signal processing problems, but at the same time possess tremendous opportunities to develop more practically useful and more computationally efficient algorithms. These challenges and opportunities will be detailed in the following section.

2. CHALLENGES AND OPPORTUNITIES

Prior to the Internet, voice communications was accomplished with the public switched telephone network (PSTN). Over the duration of a call, there exists a physical connection between the two users. In this arrangement, the earpiece at one end is used as the the speaker's extended mouth or articulator at the other end. This mode of conversation allows full-duplex and even hands-free communication with the help of a speakerphone. But because of the use of only one microphone and one loudspeaker, a sense of spatialization cannot be rendered, and the listener is unable to obtain a vivid impression of the remote speaking environment. Adding video might help, but the hearing experience is still not enjoyable for sure. Multiple microphones and loudspeakers must be employed to precisely record and faithfully reproduce the remote acoustic environment.

The television industry has witnessed a successful evolution of audio technology from now obsolete mono to prevailing stereo, and on to the highly desirable home theater with 5.1 channels. Therefore, it does not seem to be an exaggeration to believe that the multichannel mode will be the eventual technique of choice in multimedia communication systems. The transmission of real-time multichannel audio signals (possibly video signals as well) definitely consumes a larger bandwidth than before and the limited bandwidth of a traditional telephone connection will prevent us from implementing advanced multichannel communication concepts. In contrast, packet switched networks are flexible in allocating bandwidth for a particular service. With the increasing improvement of Quality-of-Service (QoS), packet switched networks will provide the needed physical connections for multimedia communications. Figure 1.1 depicts the differences between the traditional voice communication system over a circuit switched telephony network and the new multimedia communication system for telecollaboration over a packet switched network.

At the receiving room of next-generation multimedia communication systems, we aim at constructing a spatially sensible sound stage using multiple loudspeakers with object-oriented multiple-participant management. A key technical challenge at the transmitting room would be our ability to acquire high-fidelity speech while keeping speakers' spatial information with multiple microphones. In this case, we work with a complicated multiple-input multiple-output (MIMO) system. Consequently, a number of signal processing problems need to be addressed in the following broad areas: speech acquisition and enhancement; acoustic echo cancellation; sound source localization and tracking; source separation; audio coding; and realistic sound stage reproduction. In this book, we invited well-recognized experts to contribute chapters covering the state-of-the-art in the research of these focused fields.

3. ORGANIZATION OF THE BOOK

The main body of this book is organized into four parts, each of which is composed of three chapters. Part I is devoted to the speech acquisition and enhancement problem. Part II provides a detailed exposition of theory and algorithms for solving the multichannel echo cancellation problem and presents a successfully implemented real-time system. Part III concerns the source localization/tracking problem and presents a unified treatment for blind source separation of convolutive mixtures. Part IV explores audio coding and realistic sound stage reproduction.

Chapter 2 by Elko contains an updated version of a chapter that appeared in the earlier book, *Acoustic Signal Processing for Telecommunication* edited by Gay and Benesty. This new version combines the development of optimal arrays for spherical and cylindrical noise cases in a more seamless exposition.

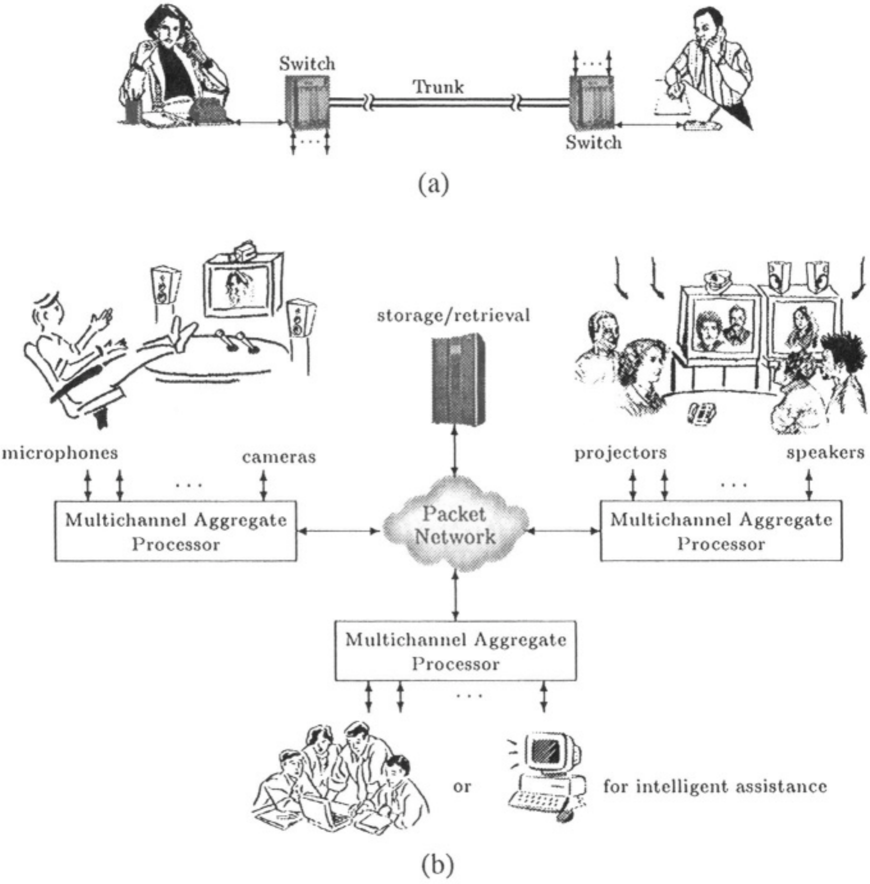

Switch
Trunk
Switch
(a)
storage/retrieval
microphones
cameras
projectors
speakers
Multichannel Aggregate Processor
Packet Network
Multichannel Aggregate Processor
Multichannel Aggregate Processor
or
for intelligent assistance
(b)

Figure 1.1 **Illustration of the difference between (a) the traditional voice communication system over a circuit switched telephony network and (b) the new multimedia communication system for telecollaboration over a packet switched network.**

The chapter covers the design and implementation of differential arrays that are by definition superdirectional in nature. Aside from their small size, one of the most beneficial features of differential arrays is the inherent independence of their directional response as a function of frequency. This quality makes them very desirable for the acoustic pickup of speech and music signals. The chapter covers several optimal differential arrays that are useful for teleconferencing and speech pickup in noisy and reverberant environments. A key issue in the design of differential arrays is their inherent sensitivity to noise and sensor mismatch. This important issue is covered in the chapter in a general way to enable one to assess the robustness of any differential array design.

Chapter 3 by Meyer and Elko describes a new spherical microphone array that enables accurate capture of the spatial components of a sound field. The array performs an orthonormal spatial decomposition of the sound pressure field using spherical harmonics. Sufficient order decomposition into these orthogonal spherical harmonic spatial modes (called eigenbeams) allows one to realize much higher spatial resolution than traditional recording systems, thereby enabling more accurate sound field capture. A general mathematical framework is given in the chapter where it is shown that these eigenbeams form the basis of a scalable representation that enables one to compute and analyze the spatial distribution of live or recorded sound fields in a computationally very efficient manner. Experimental results are shown for a real-time implementation which shows that the theory based on spherical harmonic eigenbeams matches the measured experimental data.

In many telecommunications applications, speech communications are degraded by the presence of background noise. As discussed in Chapter 4 by Diethorn, digital signal processing can be used to reduce the level of the noise for the purpose of enhancing the quality of transmitted speech. There are two main categories of approaches to noise reduction: spatial-acoustic processing methods (e.g., beamforming) and non-spatial noise suppression, or noise stripping, methods. The former category is discussed in Chapter 2 of this text. The latter category includes Wiener filtering and short-time spectral modification techniques that attempt to increase the speech-signal-to-noise ratio from knowledge of the noisy speech signal alone. Chapter 4 reviews the most popular of these methods, provides some perspective on their origin, and demonstrates a noise suppression method using actual recorded speech.

Adaptive algorithms play an important role in audio signal processing. Whenever we need to estimate and track an acoustic channel with or without a reference (input) signal, adaptive filtering is the best tool to use. In Chapter 5 by Benesty, Gänsler, Huang, and Rupp, a large number of multichannel adaptive algorithms, both in the time and frequency domains, are discussed. This discussion is developed in the context of multichannel acoustic echo cancellation where we have to identify a multiple-input multiple-output (MIMO) system (e.g., room acoustic impulse responses).

Double-talk detectors (DTDs) are vital to the operation and performance of acoustic echo cancelers. In Chapter 6 by Gänsler and Benesty, important aspects that should be considered when designing a DTD are discussed. The generic double-talk detector scheme and fundamental means for performance evaluation are discussed. A number of double-talk detectors suitable for acoustic echo cancelers are presented and objectively compared using their respective receiver operating characteristic.

In Chapter 7 by Gänsler, Fischer, Diethorn, and Benesty, the design of a real-time software acoustic echo canceler running natively under the Windows operating systems on personal computers is presented. With this software, teleconferencing is possible in wideband stereo audio over commercial IP networks in point-to-point as well as multi-point communication scenarios. The main challenge for such an implementation is to achieve sample-synchronized input and output streams for audio. This is required by the echo cancellation algorithm to maintain stable performance. Methods that achieve stable performance on hardware from various manufacturers are described. Furthermore, stereophonic echo cancellation is significantly more complicated to handle than the monophonic case because of computational complexity, nonuniqueness of solution, and convergence problems. In this chapter, the core algorithms are described including a powerful double-talk detection unit, a fast frequency-domain RLS adaptation algorithm which is optimized for non-Gaussian noise distributions, and residual echo and noise suppression. Simulation results are given which show that the algorithms achieve the theoretical bound on performance (echo attenuation). Furthermore, the software has been used for teleconferencing in wideband stereo audio over commercial IP networks.

Chapter 8 by Chen, Huang, and Benesty focuses on time delay estimation (TDE) in a reverberant environment. Particular attention is paid to the robustness of a TDE system with respect to multipath and reverberation effects. Three broad categories of approaches are studied: generalized cross-correlation, multichannel cross-correlation, and blind channel identification. The strengths and weaknesses of these approaches are elaborated and their performance (with respect to noise and reverberation) is studied and compared.

Chapter 9 by Huang, Benesty, and Elko provides an overview of fundamental concepts and a number of cutting-edge algorithms for acoustic source localization with passive microphone arrays. The localization problem is postulated from the perspective of estimation theory and the Cramèr-Rao lower bound for unbiased location estimators is derived. After an insightful review of conventional approaches ranging from maximum likelihood to least squares estimators, a recently developed linear-correction least-squares algorithm is presented that is more robust to measurement errors and that is more computationally as well as statistically efficient. At the end of Chapter 9, the design and implementation of a successful real-time acoustic source localization/tracking system for video camera steering in teleconferencing is described.

Chapter 10 by Buchner, Aichner, and Kellermann presents a unified treatment of blind source separation algorithms for convolutive mixtures as arising from reverberant acoustic environments. Based on an information-theoretical approach, time-domain and frequency-domain algorithms are described, exploiting three fundamental signal properties, as is typical, e.g., for speech and audio signals: nonwhiteness, nonstationarity, and non-Gaussianity. This framework covers existing and novel algorithms and provides both new theoretical insights and efficient practical realizations.

Chapter 11 by Schuler describes the basics of perceptual audio coding and recent developments about audio coding targeted for communications applications. It includes the fundamentals of psycho-acoustic models as they are used in today's audio coders and the principles of filter bank structures and design. Further, it describes the structure and function of standard audio coders, and explains why this structure leads to too much end-to-end delay for communications applications. Low delay audio coders and a new structure for audio coding for communications applications are then presented.

Conventional approaches to reproducing a spatially sensible sound stage are not capable of immersing a large number of listeners. Chapter 12 by Spors, Teutsch, Kuntz, and Rabenstein describes a novel technique called wave field synthesis, which is developed to overcome this problem. It is based on the principles of wave physics and is suitable for implementation with current multichannel audio hardware and software products. The listeners are not restricted in number, position, or activity and are not required to wear headphones. Furthermore, the listening area can be of arbitrary size. This chapter also addresses the problem of spatial aliasing and the compensation of non-ideal properties of loudspeakers and listening rooms, which is important for a successful implementation of wave field synthesis systems.

In Chapter 13 by Avendano, techniques based on binaural signal processing are presented that are capable of encoding and rendering sound sources accurately in three-dimensional space. In telecollaboration environments, where realistic sound-stage reproduction is a requirement for immersion, these systems offer multiple advantages. One of these is the reduced number of playback channels necessary to render the sound stage. Given that the human hearing mechanism is binaural, all spatial information available to the listener is encoded within two acoustic signals reaching the ears; thus in principle only two playback channels are necessary and sufficient to realistically render spatial sound. The decoding mechanisms that humans use to process spatial sound and the acoustic mechanisms that encode this information is described. With this knowledge, the design of virtual spatial sound (VSS) systems with applications to telecollaboration is discussed.

I

SPEECH ACQUISITION AND ENHANCEMENT

This page intentionally left blank

Chapter 2

DIFFERENTIAL MICROPHONE ARRAYS

Gary W. Elko
Avaya Labs, Avaya
gwe@avaya.com

Abstract Noise and reverberation can seriously degrade both microphone reception and loudspeaker transmission of audio signals in telecommunication systems. Directional loudspeakers and microphone arrays can be effective in combating these problems. This chapter covers the design and implementation of differential arrays that are by definition small compared to the acoustic wavelength. Differential arrays are therefore superdirectional arrays since their directivity is higher than that of a uniformly summed array with the same geometry. Aside from their small size, another beneficial feature of differential arrays is the inherent independence of their directional response as a function of frequency. Derivations are included for several optimal differential arrays that may be useful for teleconferencing and speech pickup in noisy and reverberant environments. Expressions and design details covering optimal multiple-order differential arrays are given. The results shown in this chapter should be useful in designing and selecting directional microphones for a variety of applications.

Keywords: Acoustic Arrays, Beamforming, Directional Microphones, Differential Microphones, Room Acoustics

1. INTRODUCTION

Acoustic noise and reverberation can seriously degrade both the microphone reception and the loudspeaker transmission of speech signals in communication systems. The use of small directional microphones and loudspeakers can be effective in combatting these problems. First-order differential microphones have been in existence for more than 50 years. Due to their directional (farfield) and close-talking properties (nearfield), they have proven essential for the reduction of feedback in public address systems and for communication systems in high noise and reverberant environments. In telephone applications, such as speakerphone teleconferencing, directional microphones can be very effective at reducing noise and reverberation. Since small differential arrays can offer significant improvement in teleconferencing configurations, it is expected that they will become standard components for audio communication devices in the future.

Work on various fixed and adaptive differential microphone arrays was started at the Acoustics Research Department at Bell Labs in the late 1980's. The contents of this chapter represent some of the fundamental work that was done at Bell Labs. The main focus of this chapter is to show the development of some of the necessary analytical expressions in differential microphone array design. Included are several results of potential interest to a designer of such microphone systems. Various design configurations of multiple-order differential arrays optimized under various criteria are discussed.

Generally, designs and applications of differential microphones are illustrated. Since transduction and transmission of acoustic waves are commonly reciprocal processes, the results presented here are also applicable to loudspeakers. However, differential loudspeaker implementations are problematic since large volume velocities are required to radiate sufficient acoustic power. The reasons are twofold: first, the source must be small compared to the acoustic wavelength; second, the real-part of the radiation impedance becomes very small for differential operation. Another additional factor that must be carefully accounted for in differential loudspeaker array design is the mutual radiation impedance between array elements. Treatment of these additional factors introduces analytical complexity that would significantly increase the exposition presented here. Since the goal is to introduce the essential concepts of differential arrays, the chapter focusses exclusively on the microphone array design problem.

2. DIFFERENTIAL MICROPHONE ARRAYS

The term *first-order* differential array applies to any array whose response is proportional to the combination of two components: a zero-order (acoustic pressure) signal and another proportional to the first-order spatial derivative of a scalar acoustic pressure field. Similarly, the term n^{th}*-order* differential array is used for arrays that have a response proportional to a linear combination of signal derived from spatial derivatives up to, and including n. Differential arrays have higher directivity than that of a uniformly weighted delay-sum array having the same array geometry. Arrays that have this behavior are sometimes referred to as *superdirectional* arrays. Microphone array systems that are discussed in this chapter respond to finite-differences of the acoustic pressure that closely approximate the pressure differentials for general order. Thus, the interelement spacing of the array microphones is much smaller than the acoustic wavelength and the arrays are therefore inherently superdirectional. Typically, differential arrays combine the outputs of closely-spaced microphones in an alternating sign fashion. Thus, differential arrays are also occasionally referred to as *pattern-differencing* arrays.

Before discussing various implementations of n^{th}-order finite-difference systems, expressions are developed for the n^{th}-order spatial acoustic pressure derivative in a direction **r** (the bold-type indicates a vector quantity). Since realizable differential arrays are approximations to acoustic pressure differentials, equations for general order differentials provide significant insight into the operation of these systems.

The acoustic pressure field for a propagating acoustic plane-wave can be written as

$$p(k,\mathbf{r},t) = P_o e^{j(\omega t - \mathbf{k}^T \mathbf{r})} = P_o e^{j(\omega t - kr\cos\theta)}, \qquad (2.1)$$

where P_o is the plane-wave amplitude, ω is the angular frequency, T is the transpose operator, **k** is the acoustic wavevector ($\|\mathbf{k}\| = k = \omega/c = 2\pi/\lambda$ and λ is the acoustic wavelength), c is the speed of sound, and $r = \|\mathbf{r}\|$ where **r** is the position vector relative to the selected origin. The angle θ is the angle between the position vector **r** and the wavevector **k**. Dropping the time dependence and taking the n^{th}-order spatial derivative along the direction of the position vector **r** yields:

$$\frac{d^n}{dr^n} p(k,r) = P_o(-jk\cos\theta)^n \, e^{-jkr\cos\theta}. \qquad (2.2)$$

The plane-wave solution is valid for the response to sources that are "far" from the microphone array. The term "far" implies that the distance between source and receiver is many times the square of the relevant source dimension divided by the acoustic wavelength. Using (2.2) one can conclude that the n^{th}-order differential has a bidirectional pattern component with the shape of $(\cos\theta)^n$. It can also be seen that the frequency response of a differential microphone is high-pass with a slope of $6n$ dB per octave. If the far-field assumption is relaxed, the response of the differential system to a point source located at the coordinate origin is

$$p(k,r) = P_o \frac{e^{-j(kr\cos\theta)}}{r}. \qquad (2.3)$$

The n^{th}-order spatial derivative in the radial direction **r** is

$$\frac{d^n}{dr^n} p(k,r,\theta) = P_o \frac{n!}{r^{n+1}} e^{-jkr\cos\theta} (-1)^n \sum_{m=0}^{n} \frac{(jkr\cos\theta)^m}{m!}, \qquad (2.4)$$

where r is the distance to the source. A fundamental property for differential arrays is that the general n^{th}-order array response is a weighted sum of bidirectional terms of the form $\cos^n\theta$. This property will be used in later sections. First, though, the effects of the finite-difference approximation for the spatial derivatives are investigated.

If the first-order derivative of the acoustic pressure field is expanded into a spatial Taylor series, then zero and first-order terms are required to express the general spatial derivative. The resulting equations are nothing other than finite-difference approximations to spatial derivatives. As long as the element spacing is small compared to the acoustic wavelength, the higher-order terms (namely, the higher-order derivatives) become insignificant over a desired frequency range. The resulting approximation can be expressed as the exact spatial derivative multiplied by a bias error term. For the first-order case and for a plane-wave acoustic field (2.2), the pressure derivative is

$$\frac{dp(k,r,\theta)}{dr} = -jkP_o \cos\theta \, e^{-jkr\cos\theta}. \tag{2.5}$$

A finite-difference approximation for the first-order system can be defined as

$$\begin{aligned} \frac{\Delta p(k,r,\theta)}{\Delta r} &\equiv \frac{p(k,r+d/2,\theta) - p(k,r-d/2,\theta)}{d} \\ &= \frac{-j2P_o \sin(k\,d/2 \,\cos\theta) e^{-jkr\cos\theta}}{d}, \end{aligned} \tag{2.6}$$

where d the distance between the two microphones. If we now define the amplitude bias error ϵ_1 as

$$\epsilon_1 = \frac{\Delta p/\Delta r}{dp/dr}, \tag{2.7}$$

then on-axis ($\theta = 0$),

$$\epsilon_1 = \frac{\sin kd/2}{kd/2} = \frac{\sin \pi d/\lambda}{\pi d/\lambda}. \tag{2.8}$$

Figure 2.1 shows the amplitude bias error ϵ_1 between the true pressure differential and the approximation by the difference between two closely-spaced omnidirectional (zero-order) microphones. The bias error is plotted as a nondimensional function of microphone spacing divided by the acoustic wavelength (d/λ). From Fig. 2.1, it can be seen that the element spacing must be less than 1/4 of the acoustic wavelength for the error to be less than 1 dB. Similar equations can be written for higher-order differences. The relative approximation error for these systems is low-pass in nature if the frequency range is limited to the small kd range.

In general, to realize an array that is sensitive to the n^{th} derivative of the incident acoustic pressure field, we require m p^{th}-order microphones, where, $m + p - 1 = n$. For example, a first-order differential microphone requires two zero-order microphones.

A first-order differential microphone is typically fabricated with a single movable membrane that is open to the sound field on both sides. Since the

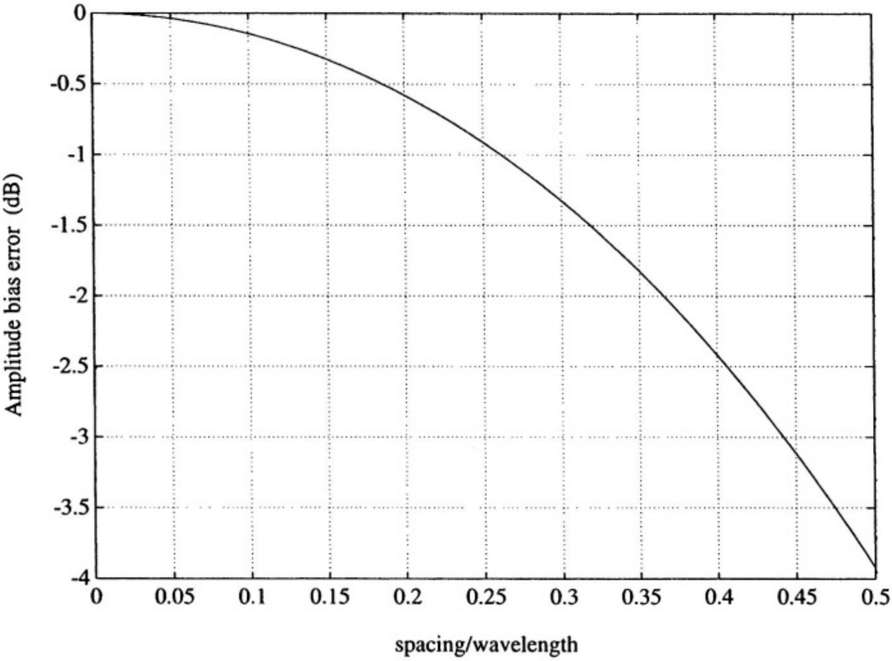


Figure 2.1 **Relative finite-difference amplitude error in dB for a plane-wave propagating along the microphone axis, as a function of element spacing divided by the acoustic wavelength.**

microphone responds to the pressure-difference across the membrane, a simple relationship to the acoustic particle velocity can be obtained from the linearized Euler equation for an ideal (no viscosity) fluid. Euler's equation can be written as

$$-\nabla p = \rho \frac{\partial \mathbf{v}}{\partial t}, \tag{2.9}$$

where ρ is the fluid density and $\mathbf{v}$ is the acoustic particle velocity. The time derivative of the particle velocity is proportional to the pressure-gradient. For an axial component of a the velocity vector, the output is proportional to the pressure differential along that axis. Thus, a first-order differential microphone is one that responds to both the scalar pressure and a component of the particle velocity of the sound field at the same measurement position. The design of higher order microphones can be formed by combinations of lower-order microphones where the sum of all of component microphone orders is equal to the desired differential microphone order.

It is instructive to analyze a first-order differential microphone to establish the basic concepts required in building and analyzing higher-order differential

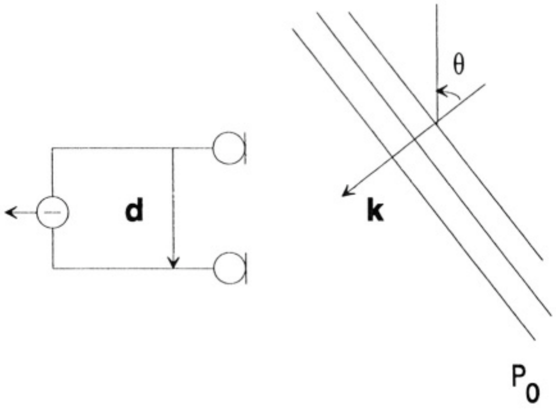


Figure 2.2 Diagram of first-order microphone composed of two zero-order (omnidirectional) microphones.

arrays. To begin, let us examine the simple two-element differential array as shown in Fig. 2.2. For a plane-wave with amplitude P_o and wavenumber k incident on a two-element array, the output can be written as

$$E_1(k,\theta) = P_o\left(1 - e^{-jkd\cos\theta}\right), \tag{2.10}$$

where d is the interelement spacing and the subscript indicates a first-order differential array. Note again that the explicit time dependence factor is neglected for the sake of compactness. If it is now assumed that the spacing is much smaller than the acoustic wavelength, then

$$E_1(k,\theta) \approx P_o kd\cos\theta. \tag{2.11}$$

As expected, the first-order array has the factor $\cos\theta$ that resolves the component of the acoustic particle velocity along the microphone axis.

By allowing the addition of time delay between these two subtracted zero-order microphones, it is possible to realize general first-order directional responses. For a plane-wave incident on this new array

$$E_1(\omega,\theta) = P_o\left(1 - e^{-j\omega(\tau + d\cos\theta/c)}\right), \tag{2.12}$$

where τ is equal to the delay applied to the signal from one microphone. For small spacing ($kd \ll \pi$ and $\omega\tau \ll \pi$),

$$E_1(\omega,\theta) \approx P_o\omega\left(\tau + d/c\cos\theta\right). \tag{2.13}$$

One thing to notice about (2.13), is that the first-order array has a high-pass frequency dependence. The term in the parentheses in (2.13) contains the array directional response. In the design of differential arrays, the array directivity function is the quantity that is of interest. To further simplify the analysis for the directivity of the first-order array, define a_0, a_1, and α_1, such that

$$\alpha_1 = a_0 = \frac{\tau}{\tau + d/c} \tag{2.14}$$

and

$$1 - \alpha_1 = a_1 = \frac{d/c}{\tau + d/c}. \tag{2.15}$$

From these definitions it can easily be seen that

$$a_0 + a_1 = 1. \tag{2.16}$$

Thus, a normalized directional response can be written as

$$E_{N_1}(\theta) = a_0 + a_1 \cos\theta = \alpha_1 + (1 - \alpha_1)\cos\theta, \tag{2.17}$$

where the subscript *N* denotes the normalized response of a first-order system, i.e., $E_{N_1}(0) = 1$. The normalization of the array response has effectively factored out the term that defines the directional response of the microphone array. The most interesting thing to notice in (2.17) is that the first-order differential array directivity function is *independent* of frequency within the region where the assumption of small spacing compared to the acoustic wavelength holds. Note that the dependent variable α_1 is itself a function of the variables d and τ.

The magnitude of (2.17) is a parametric expression for the "limaçon of Pascal" algebraic curve. The two terms in (2.17) can be seen to be the sum of a zero-order microphone (first-term) and a first-order microphone (second term), which is the general form of the first-order array. Early unidirectional microphones were actually constructed by summing the outputs of a pressure microphone and a velocity ribbon microphone (pressure-differential microphone) [12]. One implicit property of (2.17) is that for $0 \leq \alpha_1 \leq 1$, there is a maximum at $\theta = 0$ and a minimum at an angle between $\pi/2$ and π. For values of $\alpha_1 > 1/2$ the response has a minimum at 180°, although there is no null in the response. An example of the response for this case is shown in Fig. 2.3(a) and designs of this type are sometimes referred to as "subcardioid." When $\alpha_1 = 1/2$, the parametric algebraic equation has a specific form which is called a cardioid. The cardioid pattern has a null in the response at $\theta = 180°$. For values of $0 \geq \alpha_1 < 1/2$ there is a null at $90° < \theta < 180°$. Figure 2.3(b) shows a directivity response corresponding to the case where $\alpha_1 = 0.2$. For the first-order system, the solitary null is located at

$$\theta_1 = \cos^{-1}(-\frac{a_0}{a_1}) = \cos^{-1}\left(\frac{-\alpha_1}{1 - \alpha_1}\right). \tag{2.18}$$

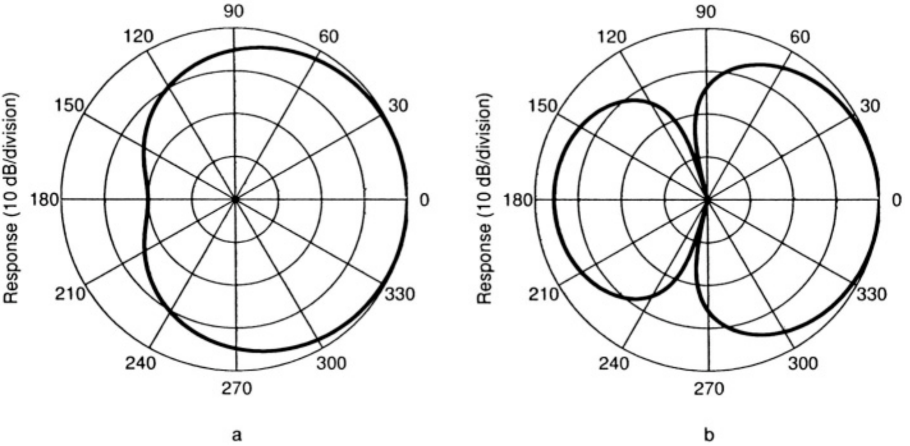


Figure 2.3 Directivity plots for first-order arrays (a) $\alpha_1 = 0.55$, (b) $\alpha_1 = 0.20$.

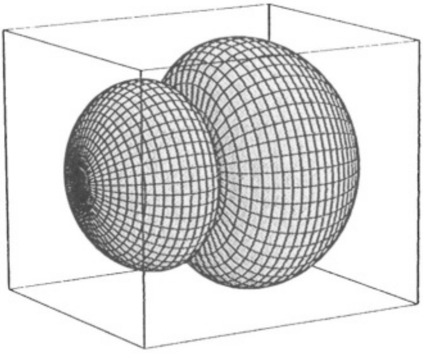


Figure 2.4 Three dimensional representation of directivity in Fig. 2.3(b). Note that the viewing angle is from the rear-half plane at an angle of approximately 225°. The viewing angle was chosen so that the rear-lobe of the array would not be obscured by the mainlobe.

Directivity patterns shown by Fig. 2.3 are actually a representation of a plane slice through the center line of the three-dimensional spherical coordinate directivity plot. Arrays discussed in this chapter are rotationally symmetric around their axes since a linear array geometry is assumed. Figure 2.4 shows a three-dimensional representation of the directivity pattern shown in Fig. 2.3(b).

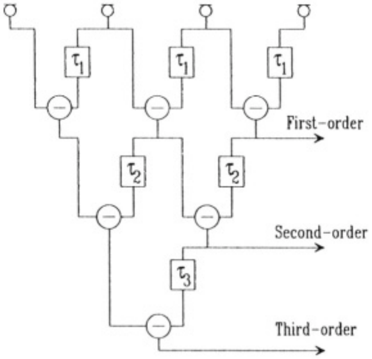


Figure 2.5 **Construction of differential arrays as first-order differential combinations up to third-order.**

A general realization of a first-order differential response is accomplished by adjusting the time delay between the two zero-order microphones that comprise the first-order system model. From (2.14) and (2.15), the value of τ determines the ratio of a_1/a_0. The value of τ is proportional to d/c, the propagation time for an acoustic wave to axially travel between the zero-order microphones. This interelement propagation time is

$$\tau = \frac{d\,a_0}{c\,a_1} = \frac{d\alpha_1}{c(1-\alpha_1)}. \tag{2.19}$$

From (2.19) and (2.18), the pattern zero is at

$$\theta_1 = \cos^{-1}\left(-\frac{c\tau}{d}\right). \tag{2.20}$$

An n^{th} order array can be written as a sum of the n^{th}-order spatial derivative and lower-order terms. An n^{th}-order array can also be written as the product of n first-order response terms as

$$E_n(\omega,\theta) = P_o \prod_{i=1}^{n} \left[1 - e^{-j\omega(\tau_i + d_i/c\cos\theta)}\right], \tag{2.21}$$

where the d_i relate to the microphone spacings, and the τ_i relate to chosen time delays. There is a design advantage in expressing the array response in terms of the products of first-order terms: it is now simple to represent higher-order systems as cascaded systems of lower order. Figure 2.5 shows how differential arrays can be constructed for up to third-order. Extension of the design technique to higher orders is straightforward. Values of τ_i can be determined by using the relationships developed in (2.14) and (2.19). The ordering of the τ_i is not important as long as $\omega\tau_i \ll \pi$.

If again it is assumed that $kd_i \ll \pi$ and $\omega\tau_i \ll \pi$, then (2.21) can be approximated as

$$E_n(\omega,\theta) \approx P_o\omega^n \prod_{i=1}^{n} (\tau_i + d_i/c \cos\theta) . \qquad (2.22)$$

Equation (2.22) can be further simplified by making the same substitution as was done in (2.14) and (2.15) for the arguments in the product term. Setting $\alpha_i = \tau_i/(\tau_i + d_i/c)$, then

$$E_n(\omega,\theta) \approx P_o\omega^n \prod_{i=1}^{n} [\alpha_i + (1 - \alpha_i)\cos\theta] . \qquad (2.23)$$

If the product in (2.23) is expanded, a power series in $\cos\theta$ can be written for the response of the n^{th}-order array to an incident plane-wave can be written as

$$E_n(\omega,\theta) = P_o A\,\omega^n \left(a_0 + a_1\cos\theta + a_2\cos^2\theta + ... + a_n \cos^n\theta\right) , \qquad (2.24)$$

where the constant *A* is an overall gain factor and we have suppressed the explicit dependence of the directivity function *E* on the variables d_i and τ_i for compactness. The only frequency dependent term in (2.24) is ω^n. Thus the frequency response of an n^{th}-order differential array can be easily compensated by a low-pass filter whose frequency response is proportional to ω^{-n}. By choosing the structure that places only a delay behind each element in a differential array, the coefficients in the power series in (2.24) are independent of frequency, resulting in an array whose beampattern is independent of frequency. To simplify the following exposition on the directional properties of differential arrays, it is assumed that the amplitude factor can be neglected. Also, since the directional pattern described by the power series in $\cos\theta$ can have any general scaling, the normalized directional response can be written as solely a function of θ,

$$E_{N_n}(\theta) = a_0 + a_1\cos\theta + a_2\cos^2\theta + ... + a_n \cos^n\theta, \qquad (2.25)$$

where the subscript *N* denotes a normalized response at $\theta = 0°$ [as in (2.17)] which implies

$$\sum_{i=0}^{n} a_i = 1. \qquad (2.26)$$

In general, n^{th}-order differential microphones have at most, n nulls (zeros). This follows directly from (2.25) and the fundamental theorem of algebra. Equation (2.25) can also be written in "canonic" form as the product of first-order terms

$$E_{N_n}(\theta) = \prod_{i=1}^{n} [\alpha_i + (1 - \alpha_i)\cos\theta] . \qquad (2.27)$$

Note that the frequency dependent variable ω in (2.25) and (2.27) has been dropped since it was shown that the frequency response for small interelement spacing is simply proportional to ω^n. The terms a_i in (2.25) can take on any desired value by adjusting the delays used in defining the desired differential microphone array. For the second-order array

$$E_{N_2}(\theta) = a_0 + a_1 \cos\theta + a_2 \cos^2\theta. \tag{2.28}$$

Equation (2.28) can also be factored into two first-order terms and written as

$$E_{N_2}(\theta) = [\alpha_1 + (1-\alpha_1)\cos\theta][\alpha_2 + (1-\alpha_2)\cos\theta], \tag{2.29}$$

where

$$\begin{aligned} a_0 &= \alpha_1\alpha_2, \\ a_1 &= \alpha_1(1-\alpha_2) + \alpha_2(1-\alpha_1), \\ a_2 &= (1-\alpha_1)(1-\alpha_2), \end{aligned} \tag{2.30}$$

or

$$\begin{aligned} \alpha_1 &= a_0 + a_1/2 \pm \sqrt{(a_0 + a_1/2)^2 - a_0}, \\ \alpha_2 &= a_0 + a_1/2 \mp \sqrt{(a_0 + a_1/2)^2 - a_0}. \end{aligned} \tag{2.31}$$

As shown, the general form of the second-order system is the sum of second-order, first-order and zero-order terms. If certain constraints are placed on the values of a_0 and a_1, it can be seen that there are two nulls (zeros) in the interval $0 \leq \theta < \pi$. The array response pattern is symmetric about $\theta = 0$. These zeros can be explicitly found at the angles θ_1 and θ_2:

$$\theta_1 = \cos^{-1}\left(\frac{-\alpha_1}{1-\alpha_1}\right), \tag{2.32}$$

$$\theta_2 = \cos^{-1}\left(\frac{-\alpha_2}{1-\alpha_2}\right), \tag{2.33}$$

where now α_1 and α_2 can take on either positive or negative values. If the resulting beampattern is constrained to have a maximum at $\theta = 0°$, then the values of α_1 and α_2 can only take on certain values; we have ruled out designs that have a higher sensitivity at any angle other than $\theta = 0°$. An interesting thing to note is that negative values of α_1 or α_2 correspond to a null moving into the front half-plane. Negative values of α_1 for the first-order microphone can be shown to have a rear-lobe sensitivity that exceeds the sensitivity at 0°. Since (2.29) is the product of two first-order terms, emphasis of the rear-lobe caused by a negative value of α_2, can be counteracted by the zero from the term containing α_1. As a result, a beam-pattern can be found for the second-order microphone that has maximum sensitivity at $\theta = 0°$ and a null in the front-half plane. This result also implies that the beamwidth of a second-order microphone with a negative value of α_2 is narrower than that of the second-order dipole ($\cos^2(\theta)$ directional dependence).

It is straightforward to extend the previous results to the third-order and higher-order cases. For completeness, the equation governing the directional characteristics for the third-order array is

$$E_{N_3}(\theta) = a_0 + a_1 \cos\theta + a_2 \cos^2\theta + a_3 \cos^3\theta. \tag{2.34}$$

If certain constraints are placed on the coefficients in (2.34), it can be factored into three real roots:

$$E_{N_3}(\theta) = [\alpha_1 + (1 - \alpha_1)\cos\theta][\alpha_2 + (1 - \alpha_2)\cos\theta][\alpha_3 + (1 - \alpha_3)\cos\theta]. \tag{2.35}$$

Third-order microphones have the possibility of three zeros that can be placed at desired locations in the directivity pattern. Solving this cubic equation yields expressions for α_1, α_2, and α_3 in terms of a_0, a_1, a_2, and a_3. However, these expressions are algebraically cumbersome and not repeated here.

3. ARRAY DIRECTIONAL GAIN

In order to "best" reject the noise in an acoustic field, one needs to optimize how multiple microphones are linearly combined. Specifically we need to consider the directional gain, i.e. the gain of the microphone array in a noise field over that of a simple omnidirectional microphone. A common quantity used is the directivity factor *Q*, or equivalently, the directivity index *DI* $[10\log_{10}(Q)]$. The standard definition for the directivity index is computed for general three-dimensional isotropic sound fields. However, there are possible conditions in room acoustics where a sound field can be reasonably modeled as a two-dimensional or cylindrical sound field. To logically denote equations for spherical and cylindrical fields, a subscript of *C* is used to denote the cylindrical two-dimensional case.

A general expression for the directivity factor can be defined as

$$Q(\omega, \theta_0, \phi_0) = \frac{4\pi \mid E(\omega, \theta_0, \phi_0) \mid^2}{\int_0^{2\pi} \int_0^{\pi} \mid E(\omega, \theta, \phi) \mid^2 u(\omega, \theta, \phi) \sin\theta d\theta d\phi}, \tag{2.36}$$

where the angles θ and ϕ are the standard spherical coordinate angles, θ_0 and ϕ_0 are the angles at which the directivity factor is being measured, $E(\omega, \theta, \phi)$ is the pressure response of the array, and $u(\omega, \theta, \phi)$ is the distribution of the noise power. The function u is normalized such that

$$\frac{1}{4\pi} \int_0^{2\pi} \int_0^{\pi} u(\omega, \theta, \phi) \sin\theta d\theta\, d\phi = 1. \tag{2.37}$$

For a cylindrical sound field, the equation for a general directivity factor can be written as

$$Q_C(\omega,\phi_o) = \frac{2\pi \mid E(\omega,\phi_o) \mid^2}{\int_0^{2\pi} \mid E(\omega,\phi) \mid^2 u(\omega,\phi)d\phi}. \tag{2.38}$$

Similarly, the distribution of the noise power is defined as

$$\frac{1}{2\pi}\int_0^{2\pi} u_C(\omega,\phi)\,d\phi = 1. \tag{2.39}$$

In general, the directivity factor Q can be written as the ratio of two Hermitian quadratic forms [3] as

$$Q = \frac{\mathbf{w}^{\mathcal{H}}\mathbf{A}\mathbf{w}}{\mathbf{w}^{\mathcal{H}}\mathbf{B}\mathbf{w}}, \tag{2.40}$$

where

$$\mathbf{A} = \mathbf{S}_0\mathbf{S}_0^{\mathcal{H}}, \tag{2.41}$$

w is the complex weighting applied to the microphones and $\mathcal{H}$ represents the complex conjugate transpose. Elements of the matrix **B** are defined as

$$b_{mn} = \frac{1}{4\pi}\int_0^{2\pi}\int_0^{\pi} u(\omega,\theta,\phi)\exp\left[j\mathbf{k}^T(\mathbf{r}_m - \mathbf{r}_n)\right]\sin\theta\,d\phi\,d\theta, \tag{2.42}$$

where **r** are the microphone element position vectors. For the cylindrical sound field case,

$$b_{Cmn} = \frac{1}{2\pi}\int_0^{2\pi} u_C(\omega,\phi)\exp\left[j\mathbf{k}^T(\mathbf{r}_m - \mathbf{r}_n)\right]d\phi. \tag{2.43}$$

The elements of the vector $\mathbf{S}_0$ are defined as

$$s_{0n} = \exp(j\mathbf{k}_0^T\mathbf{r}_n). \tag{2.44}$$

Note that for clarity the explicit functional dependencies of the above equations on the angular frequency ω has been omitted. The solution for the maximum of Q, which is a Rayleigh quotient, is obtained by finding the maximum generalized eigenvector of the homogeneous equation

$$\mathbf{A}\mathbf{w} = \lambda_M\mathbf{B}\mathbf{w}. \tag{2.45}$$

The maximum eigenvalue of (2.45) is given by

$$\lambda_M = \mathbf{S}_0^{\mathcal{H}}\mathbf{B}^{-1}\mathbf{S}_0. \tag{2.46}$$

The corresponding eigenvector contains the weights for combining the elements to obtain the maximum directional gain

$$\mathbf{w}_{opt} = \mathbf{B}^{-1}\mathbf{S}_0. \tag{2.47}$$

This result states the general principle of "whiten and match," where the matrix inverse $\mathbf{B}$ is the spatial whitening filter and the steering vector $\mathbf{S}_0$ is the matching to signals propagating from the source direction. In general, the optimal weights $\mathbf{w}_{opt}$ are a function of frequency, array geometry, element directivity and the spatial distribution of the noise field.

4. OPTIMAL ARRAYS FOR ISOTROPIC FIELDS

Acoustic reverberation in rooms has historically been modeled as spherically isotropic noise. A spherically isotropic noise field can be constructed by combining uncorrelated noises propagating in all directions with equal power. In room acoustics this modeled noise field is referred to as a "diffuse" sound field and has been the model used for many investigations into the statistical distributions of reverberant sound pressure fields. Although the standard to model for reverberant acoustic fields has been the "diffuse" field model (spherically isotropic noise), another noise field that is appropriate for room acoustics is cylindrical noise. In many rooms, where carpet and ceiling tiles are used, the main surfaces of absorption are the ceiling and the floor. As a result, a cylindrical noise field model that has the noise propagating in the axial plane may be more appropriate. Since it is of interest to design microphone systems that optimally reject reverberant sound fields, the optimization of array gain will be given for both spherically and cylindrically "diffuse" sound fields.

4.1 MAXIMUM DIRECTIONAL GAIN

Publications on the maximization of the directional gain for an arbitrary array have been quite extensive [17, 18, 13, 4, 16, 8]. Uzkov [17] showed for uniformly spaced omnidirectional microphones that the directional gain reaches N^2 as the spacing between the elements goes to zero. A maximum value of directional gain is obtained when a collinear array is used in end-fire operation. Weston [18] has shown the same result by an alternative method. Parsons [13] has extended the proof to include nonuniformly spaced arrays that are much smaller than the acoustic wavelength. A proof given here also relies on the assumption that the elements are closely-spaced compared to the acoustic wavelength. The approach taken here is similar to that of Chu [4], Tai [16], and Harrington [8], who expanded the radiated and received field in terms of spherical wave functions. Chu did not look explicitly at the limiting case. Harrington did examine the limiting case of vanishing spacing, but his analysis involved approximations that are not necessary in the following analysis.

4.1.1 Spherically Isotropic Fields. For a spherically isotropic field and for omnidirectional microphone elements,

$$u(\omega,\theta,\phi) = 1. \tag{2.48}$$

In general, the directivity of N closely-spaced microphones can be expanded in terms of spherical wave functions. Now let us express the farfield pressure response $E(\theta,\phi)$ as a summation of orthogonal Legendre polynomials,

$$E(\theta,\phi)=\sum_{n=0}^{N-1}\sum_{m=0}^{n} h_{nm}P_n^m[\cos(\theta-\theta_z)]\cos m(\phi-\phi_z), \tag{2.49}$$

where the sum has been limited to the number of degrees of freedom in the N-element microphone case, where the P_n^m are the associated Legendre functions, and θ_z and ϕ_z are possible rotations of the coordinate system. Now define

$$G_{nm}(\theta,\phi)=P_n^m[\cos(\theta-\theta_z)]\cos\, m(\phi-\phi_z). \tag{2.50}$$

The normalization of the function G_{nm} is

$$\begin{aligned} N_{nm} &= \int_0^{2\pi}\cos^2 m\phi\int_{-1}^{1}[P_n^m(\eta)]^2\, d\eta d\phi \\ &= \frac{4\pi(n+m)!}{\varepsilon_m(2n+1)(n-m)!}, \end{aligned} \tag{2.51}$$

where the function ε_m is defined as

$$\varepsilon_m=\begin{cases}1 & m=0\\ 2 & m>0\end{cases}. \tag{2.52}$$

By using the orthogonal Legendre function expansion expressions for the directivity factor can be written as

$$Q(\theta_o,\phi_o)=4\pi\frac{\left[\sum_{n=0}^{N-1}\sum_{m=0}^{n}h_{nm}G_{nm}(\theta_o,\phi_o)\right]^2}{\sum_{n=0}^{N-1}\sum_{m=0}^{n}h_{nm}^2N_{nm}}. \tag{2.53}$$

To find the maximum of (2.53), we set the derivative of Q with respect to h_{nm} for all n and m to zero. The resulting maximum occurs when

$$\begin{aligned} Q_{max} &= \sum_{n=0}^{N-1}\sum_{m=0}^{n}\frac{4\pi\left[G_{nm}(\theta_o,\phi_o)\right]^2}{N_{nm}} \\ &= \sum_{n=0}^{N-1}\sum_{m=0}^{n}\frac{4\pi\left[P_n^m(\cos(\theta_o-\theta_z))\cos\, m(\phi_o-\phi_z)\right]^2}{N_{nm}} \\ &\le \sum_{n=0}^{N-1}\sum_{m=0}^{n}\frac{4\pi\left[P_n^m(\cos(\theta_o-\theta_z))\right]^2}{N_{nm}}. \end{aligned} \tag{2.54}$$

The inequality in (2.54) infers that the maximum must occur when $\phi_o = \phi_z$. This result can be seen by using the addition theorem of Legendre polynomials,

$$P_n(\cos\psi) = \sum_{m=0}^{n} \varepsilon_m \frac{(n-m)!}{(n+m)!} P_n^m(\cos\theta_o) P_n^m(\cos\theta_z)\cos(m[\phi_o - \phi_z]). \tag{2.55}$$

The angle subtended by two points on a sphere is ψ and is expressed in terms of spherical coordinates as

$$\cos\psi = \cos\theta_o\cos\theta_z + \sin\theta_o\sin\theta_z\cos(\phi_o - \phi_z). \tag{2.56}$$

Equation (2.55) maximizes for all n, when $\psi = 0$. Therefore (2.54) maximizes when $\theta_o = \theta_z$. Since

$$P_n^m(1) = \begin{cases} 1 & m = 0 \\ 0 & m > 0 \end{cases}, \tag{2.57}$$

(2.54) can be reduced to

$$\begin{aligned} Q_{max} &= \sum_{n=0}^{N-1} (2n+1) \\ &= N^2. \end{aligned} \tag{2.58}$$

Thus, the maximum directivity factor Q for N closely-spaced omnidirectional microphones is N^2. A proof of the maximum directivity factor for general spacing does not appear to be tractable, however it is believed that the limit established in (2.58) is true for general microphone spacing.

4.1.2 Cylindrically Isotropic Fields. For a cylindrically isotropic field we have uncorrelated plane waves arriving with equal probability from any angle wave vector direction that lies in the ϕ plane. The cylindrical directivity factor for this field is therefore defined as

$$Q_C(\omega, \phi_o) = \frac{|\, E(\omega, \phi_o)\,|^2}{\frac{1}{2\pi}\int_0^{2\pi} |\, E(\omega,\phi)\,|^2\, u(\omega,\phi) d\phi}. \tag{2.59}$$

Again, the general weighting function u allows for the possibility of a nonuniform distribution of noise power and microphone element directivity. For isotropic noise fields and omnidirectional microphones,

$$u(\omega, \phi) = 1. \tag{2.60}$$

Following the development in the last section, we expand the directional response of N closely-spaced elements in a series of orthogonal cosine functions.

For cylindrical fields we can use the normal expansion in the ϕ dimension:

$$E(\phi) = \sum_{m=0}^{N-1} h_m \cos(m[\phi - \phi_z]). \tag{2.61}$$

The normalization of these cosine functions is simply [1]:

$$\begin{aligned} N_m &= \int_0^{2\pi} \cos^2(m\phi)\, d\phi \\ &= \frac{2\pi}{\varepsilon_m}. \end{aligned} \tag{2.62}$$

For cylindrical fields, the directivity factor can therefore be written as

$$Q_C(\phi_o) = \frac{2\pi \left[\sum_{m=0}^{N-1} h_m \cos(m[\phi_o - \phi_z])\right]^2}{\sum_{m=0}^{N-1} h_m^2 N_m}. \tag{2.63}$$

The maximum is found by equating the derivative of this equation for Q_C with respect to the h_m weights to zero. The result is

$$Q_{Cmax}(\phi_o) = \sum_{m=0}^{N-1} \frac{\cos^2(m[\phi_o - \phi_z])}{N_m}. \tag{2.64}$$

The equation for Q_C maximizes when $\phi_o = \phi_z$. Therefore

$$\begin{aligned} Q_{Cmax} &= \sum_{m=0}^{N-1} \varepsilon_m \\ &= 2N - 1. \end{aligned} \tag{2.65}$$

The above result indicates that the maximum directivity factor of N closely-spaced omnidirectional microphones in a cylindrically correlated sound field is $2N - 1$. This result is apparently known in the microphone array processing community [5], but apparently a general proof that has not been published. One obvious conclusion that can be drawn from the above result is that the rate of increase in directivity factor as a function of the number of microphones is much slower for a cylindrically isotropic field than a spherically isotropic field.

A plot comparing the maximum gain for microphone arrays containing up to ten elements for both spherically and cylindrically isotropic fields is shown in Fig. 2.6. There are two main trends that can easily be seen in Fig. 2.6. First, the gain in directivity index decreases as the number of elements (order) is increase. Second, the difference between the maximum gains for spherical and cylindrical fields is quite sizable and increases as the number of elements is increased.

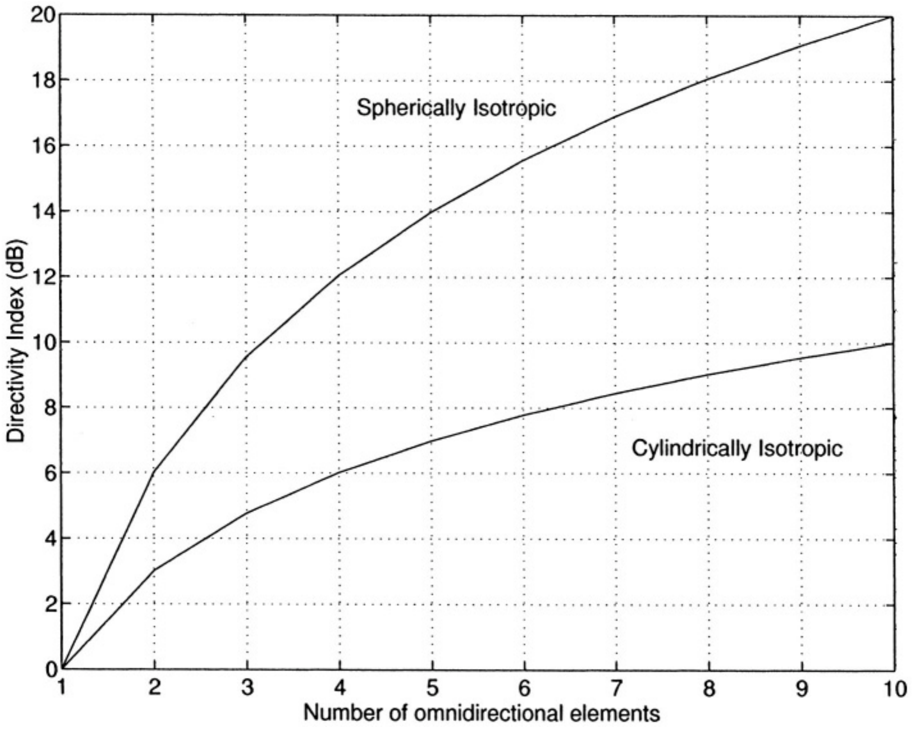


Figure 2.6 Maximum gain of an array of N omnidirectional microphones for spherical and cylindrical isotropic noise fields.

The first observation is not too problematic since practical differential arrays implementation are limited to third-order. The second observation shows that attainable gain in cylindrical fields might not result in required microphone array gains for desired rejection of noise and reverberation in rooms that have low absorption in the axial plane.

4.2 MAXIMUM DIRECTIVITY INDEX FOR DIFFERENTIAL MICROPHONES

As was shown in Section 2, there are an infinite number of possibilities for differential array designs. What is of interest here are the special cases that are optimal in some respect. For microphones that are axisymmetric, as is the case for all of the microphones covered here, (2.36) can be written in a simpler form:

$$Q(\omega) = \frac{2}{\int_0^\pi \mid E_N(\omega, \theta, \phi) \mid^2 \sin\theta d\theta}, \tag{2.66}$$

and for the cylindrical field case,

$$Q_C(\omega) = \frac{2\pi}{\int_0^{2\pi} \mid E_N(\omega,\phi) \mid^2 d\phi}, \tag{2.67}$$

where it has also been assumed that the directions θ_0, ϕ_0 are in the direction of maximum sensitivity and that the array sensitivity function is normalized: $\mid E_N(\omega, \theta_0, \phi_0) \mid = 1$. If we now insert the formula from (2.25) and carry out the integration, we find the directivity factor expressed in terms of a_j as

$$Q(a_0, ..., a_n) = \left[\sum_{i=0}^{n} \sum_{\substack{j=0 \\ i+j \text{ even}}}^{n} \frac{a_i a_j}{1+i+j} \right]^{-1} . \tag{2.68}$$

The directivity factor for a general n^{th}-order differential array (no normalization assumption) can be written as

$$Q(a_0, ..., a_n) = \left[\sum_{i=0}^{n} a_i\right]^2 \left[\sum_{i=0}^{n} \sum_{\substack{j=0 \\ i+j \text{ even}}}^{n} \frac{a_i a_j}{1+i+j} \right]^{-1} = \frac{\mathbf{a}^T\mathbf{Ba}}{\mathbf{a}^T\mathbf{Ha}}, \tag{2.69}$$

similarly, the cylindrical case can be written as

$$Q_C(a_0, ..., a_n) = \frac{\mathbf{a}^T\mathbf{Ba}}{\mathbf{a}^T\mathbf{H}_C\mathbf{a}}, \tag{2.70}$$

Where **H** and $\mathbf{H}_C$ are Hankel matrices given by

$$H_{i,j} = \begin{cases} \frac{1}{1+i+j} & \text{if i+j even} \\ 0 & \text{otherwise} \end{cases}$$

and

$$H_C i, j = \begin{cases} \frac{(i+j-1)!!}{(i+j)!!} & \text{if i+j even} \\ 0 & \text{otherwise} \end{cases} .$$

The vector **a** is defined as

$$\mathbf{a}^T = \{a_0, a_1, ..., a_n\} \tag{2.71}$$

Table 2.1 Table of maximum array gain Q, and corresponding eigenvector for differential arrays from first to fourth-order for spherically isotropic noise fields.

order	max eigenvalue	eigenvector
1	4	[1/4 3/4]
2	9	[-1/6 1/3 5/6]
3	16	[-3/32 -15/32 15/32 35/32]
4	25	[0.075 -0.300 -1.050 0.700 1.575]

and the matrix **B** is

$$\mathbf{B} = \mathbf{b}\mathbf{b}^T, \tag{2.72}$$

where

$$\mathbf{b}^T = \overbrace{\{1, 1, ..., 1\}}^{n+1}. \tag{2.73}$$

From (2.69) we can see that the directivity factors Q and Q_C, are both Rayleigh quotients for two hermitian forms. From Section 3, the maximum of the Rayleigh quotient is reached at a value equal to the largest generalized eigenvalue of the equivalent generalized eigenvalue problem,

$$\mathbf{Bx} = \lambda \mathbf{Hx}, \tag{2.74}$$

where again, λ is the general eigenvalue and $\mathbf{x}$ is the corresponding general eigenvector. The eigenvector corresponding to the largest eigenvalue will contain the coefficients a_i which maximize the directivity factor Q and Q_C. Since **B** is a dyadic product there is only one eigenvector $\mathbf{x} = \mathbf{H}^{-1}\mathbf{b}$ with the eigenvalue $\mathbf{b}^T\mathbf{H}^{-1}\mathbf{b}$. Thus

$$\max_{\mathbf{a}} Q \;=\; \lambda_m = \mathbf{b}^T\mathbf{H}^{-1}\mathbf{b}, \tag{2.75}$$

and similarly,

$$\max_{\mathbf{a}} Q_C \;=\; \lambda_m = \mathbf{b}^T\mathbf{H}_C^{-1}\mathbf{b}. \tag{2.76}$$

Tables 2.1 and 2.2 give the maximum array gain (largest eigenvalue) and corresponding eigenvectors (values of a_j that maximize the directivity factor), for differential orders up to fourth-order. Note that the largest eigenvector has been scaled such that the microphone output is unity at $\theta = 0°$. Figure 2.7 contains plots of the directivity patterns for the differential arrays given in Table 2.1.

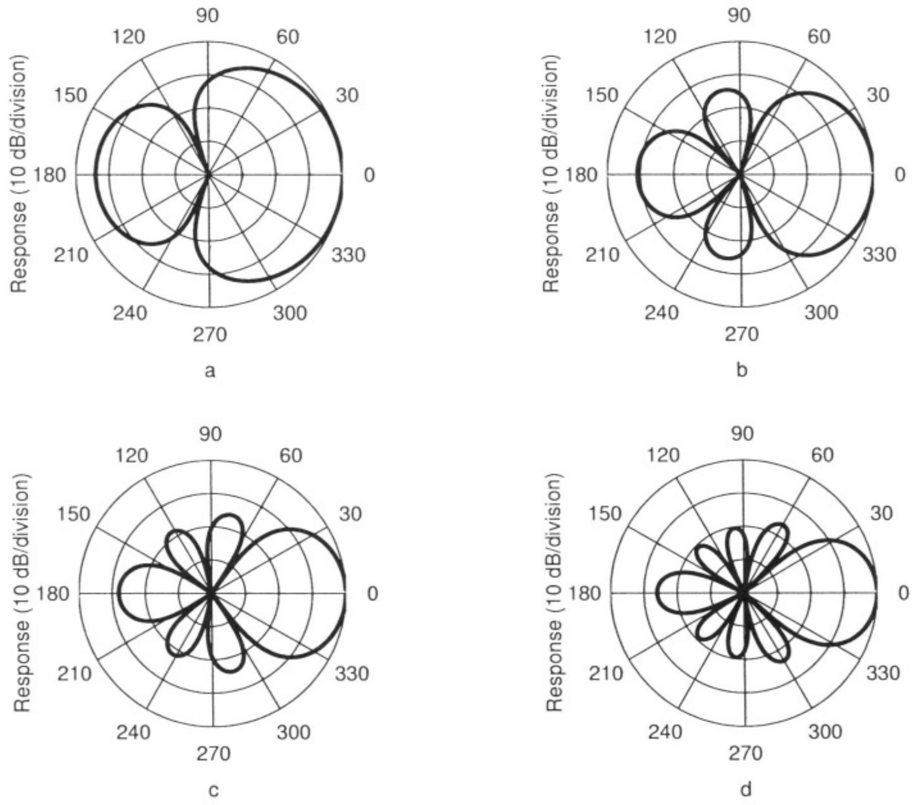


Figure 2.7 Optimum directivity patterns for differential arrays in a spherically isotropic noise field for (a) first, (b) second, (c) third, and (d) fourth-order.

Table 2.2 Table of maximum eigenvalue and corresponding eigenvector for differential arrays from first to fourth-order, for cylindrically isotropic noise fields.

order	max eigenvalue	eigenvector
1	3	[1/3 1/2]
2	5	[-1/5 2/5 4/5]
3	7	[-1/7 -4/7 4/7 8/7]
4	9	[1/9 -4/9 -4/3 8/9 16/9]

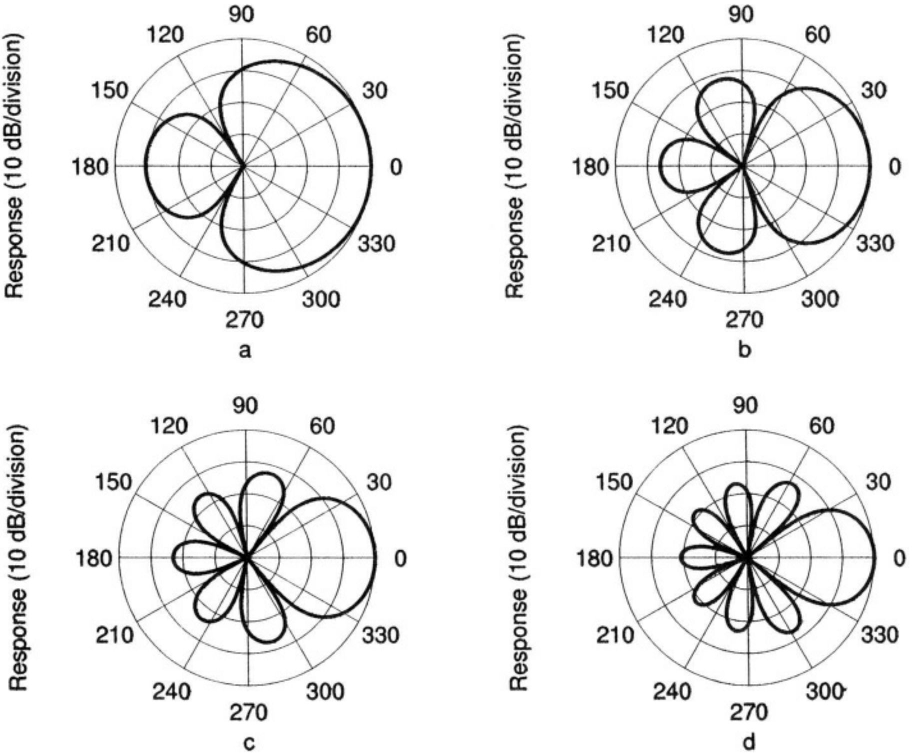


Figure 2.8 Optimum directivity patterns for differential arrays in a cylindrically isotropic noise field for (a) first, (b) second, (c) third, and (d) fourth-order.

Corresponding plots of the highest array gain directivity patterns for cylindrical sound fields in Table 2.2 are shown in Fig. 2.8.

The directivity index $(10 \log_{10} Q)$is an extremely useful measure in quantifying the directional properties of microphones and loudspeakers. It provides a rough estimate of the relative gain in signal-to-reverberation for a directional microphone in a diffuse reverberant environment. However, the directivity index might be misleading with respect to the performance of directional microphones in non-diffuse fields.

4.3 MAXIMUM FRONT-TO-BACK RATIO

Another possible measure for the "merit" of an array is the front-to-back rejection ratio, i.e., the directional gain of the microphone for signals propagating to the front of the microphone relative to signals propagating to the rear. One such quantity was suggested by Marshall and Harry [11] which will be referred to here as "*F*," for the front-to-back ratio. The ratio *F* is defined as

$$F(\omega) = \frac{\int_0^{2\pi} \int_0^{\pi/2} \mid E(\omega, \theta, \phi) \mid^2 \sin\theta d\theta d\phi}{\int_0^{2\pi} \int_{\pi/2}^{\pi} \mid E(\omega, \theta, \phi) \mid^2 \sin\theta d\theta d\phi}, \tag{2.77}$$

where the angles θ and ϕ are again the spherical coordinate angles and E(ω,θ, ϕ) is the far-field pressure response. For a cylindrical field, the front-to-back ratio can similarly be written as

$$F_C = \frac{\int_0^{\pi/2} \mid E(\phi) \mid^2 \, d\phi}{\int_{\pi/2}^{\pi} \mid E(\phi) \mid^2 \, d\phi}. \tag{2.78}$$

For axisymmetric microphones (2.77) can be written in a simpler form by uniform integration over ϕ,

$$F(\omega) = \frac{\int_0^{\pi/2} \mid E_N(\omega, \theta, \phi) \mid^2 \sin\theta d\theta}{\int_{\pi/2}^{\pi} \mid E_N(\omega, \theta, \phi) \mid^2 \sin\theta d\theta}. \tag{2.79}$$

Carrying out the integration of (2.79) and using the form of (2.25) yields the front-to-rear power ratio in terms of the weighting vector **a**,

$$\begin{aligned} F(a_0, \ldots, a_n) &= \left[\sum_{i=0}^{n} \sum_{j=0}^{n} \frac{a_i a_j}{1+i+j} \right] \left[\sum_{i=0}^{n} \sum_{j=0}^{n} \frac{(-1)^{i+j} a_i a_j}{1+i+j} \right]^{-1} \\ &= \frac{\mathbf{a}^T \mathbf{B} \mathbf{a}}{\mathbf{a}^T \mathbf{H} \mathbf{a}}, \end{aligned} \tag{2.80}$$

where **H** is a Hankel matrix given by

$$H_{i,j} = \frac{(-1)^{i+j}}{1+i+j}. \tag{2.81}$$

B is a special form of a Hankel matrix designated as a Hilbert matrix and is given by

$$B_{i,j} = \frac{1}{1+i+j}. \tag{2.82}$$

Similarly, for the cylindrical case,

$$F_C = \frac{\mathbf{a}^T \mathbf{B}_C \mathbf{a}}{\mathbf{a}^T \mathbf{H}_C \mathbf{a}},$$

where

$$B_{Cij} = \frac{\Gamma(\frac{1+i+j}{2})}{\Gamma(\frac{2+i+j}{2})},$$

Table 2.3 Table of maximum F ratio and corresponding eigenvector for differential arrays from first to fourth-order for spherically isotropic noise fields.

order	max eigenvalue	eigenvector
1	$7+4\sqrt{3}$	$[\frac{1}{1+\sqrt{3}}\ \frac{1}{1+\sqrt{3}}]$
2	$127+48\sqrt{7}$	$[\frac{1}{2(3+\sqrt{7})}\ \frac{\sqrt{7}}{3+\sqrt{7}}\ \frac{5}{2(3+\sqrt{7})}]$
3	$\approx$5875	$\approx$ [0.0184 0.2004 0.4750 0.3061]
4	$\approx$151695	[0.0036 0.0670 0.2870 0.4318 0.2107]

Table 2.4 Table of maximum eigenvalue corresponding to the maximum front-to-back ratio and corresponding eigenvector for differential arrays from first to fourth-order, for cylindrically isotropic noise fields.

order	max eigenvalue	eigenvector
1	$7+4\sqrt{3}$	[$\sqrt{2}$-1 2-$\sqrt{2}$]
2	$\frac{9\pi^2+12\sqrt{22}\pi+88}{9\pi^2-88}$	$\approx$ [0.103 0.484 0.413]
3	$\approx$ 11556	$\approx$[0.002 0.217 0.475 0.286]
4	$\approx$ 336035	$\approx$ [0.00430 0.07429 0.29914 0.42521 0.19705]

and

$$H_{Cij} = (-1)^{i+j}\frac{\Gamma(\frac{1+i+j}{2})}{\Gamma(\frac{2+i+j}{2})}, \tag{2.83}$$

where Γ is the Gamma function [1]. The matrices $\mathbf{H}, \mathbf{H}_C$, $\mathbf{B}$, and $\mathbf{B}_C$ are real Hankel matrices and are positive definite. The resulting eigenvalues are therefore positive real numbers and the eigenvectors are real. Tables 2.3 and 2.4 summarize the results for the maximum front-to-back ratios for differential arrays up to fourth-order. The maximum eigenvalue for the third and fourth-order cases result in very complex algebraic expressions and only numeric results are given. Plots showing the highest front-to-back power ratio directivity patterns given by the optimum weights in Tables 2.3-2.4 are displayed in Figs. 2.9-2.10. One observation that can be made by comparing the cylindrical and spherical noise results is that the directivity patterns are fairly similar. The differences are due to the lack of the sine term in the denominator of (2.36). The optimal patterns for cylindrically isotropic fields have smaller sidelobes in the rear-half

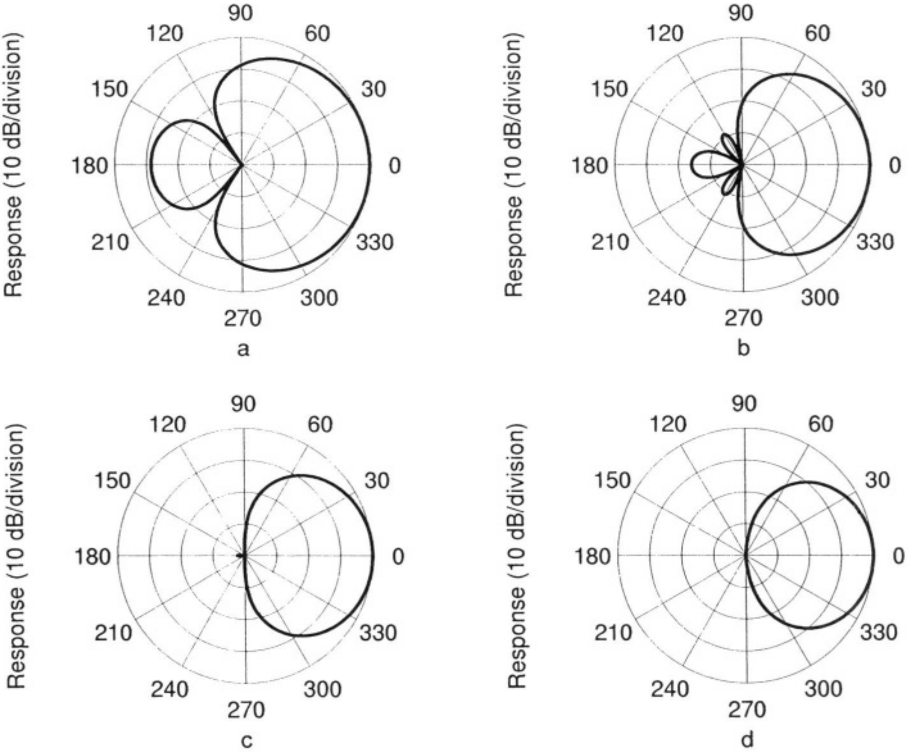


Figure 2.9 **Directivity patterns for maximum front-to-back power ratio for differential arrays in a spherically isotropic noise field for (a) first, (b) second, (c) third, and (d) fourth-order.**

of the microphone since this area is not weighted down by the sine term. Table 2.5 summarizes the optimal microphone designs presented in this section. The table also includes columns for the 3 dB beamwidth and the position of the pattern nulls. Knowledge of the null positions for different array designs allows one to easily realize any higher-order array by combining first-order sections in a tree architecture as shown in Fig 2.5.

The results summarized in Table 2.5 also show that there is relatively small difference between the optimal designs of differential arrays for the spherical and cylindrical isotropic noise fields. Typical differences between directional gains for either cylindrical or spherical isotropy assumptions are less than a few tenths of a dB; most likely an insignificant amount. Probably the most important detail to notice is that the rate of increase in the directional gain versus differential array order is much smaller for cylindrically isotropic fields. This conclusion was also shown earlier (see Fig. 2.6).

Table 2.5 Table of maximum directional gain and front-to-back power ratio for differential arrays from first to fourth-order, for cylindrically and spherically isotropic noise fields.

Mic. order	DI_C dB	F_C dB	DI dB	F dB	Beamwidth degs	Null(s) degs
Maximum gain for cylindrical noise						
1st	4.8	10.9	5.9	11.1	112°	120
2nd	7.0	10.9	9.4	7.5	65°	72,144
3rd	8.5	13.9	11.8	10.3	46°	51,103,154
4th	9.5	13.9	13.7	8.9	36°	80,120,160
Maximum gain for spherical noise						
1st	4.6	7.4	6.0	8.5	105°	109
2nd	6.9	9.7	9.5	8.5	65°	73,134
3rd	8.3	12.4	12.0	11.2	48°	55,100,145
4th	9.4	13.8	14.0	11.2	38°	44,80,117,152
Maximum front-to-back ratio for cylindrical noise						
1st	4.6	12.8	5.4	10.9	120°	135
2nd	6.3	26.3	8.2	23.4	81°	106,153
3rd	7.2	40.6	9.8	37.0	66°	98,125,161
4th	7.8	55.3	10.9	51.1	57°	95,112,137,165
Maximum front-to-back ratio for spherical noise						
1st	4.8	12.0	5.7	11.4	115°	125
2nd	6.4	25.1	8.3	24.0	80°	104,144
3rd	7.2	39.2	9.9	37.7	65°	97,122,154
4th	7.8	53.6	11.0	51.8	57°	94,111,133,159

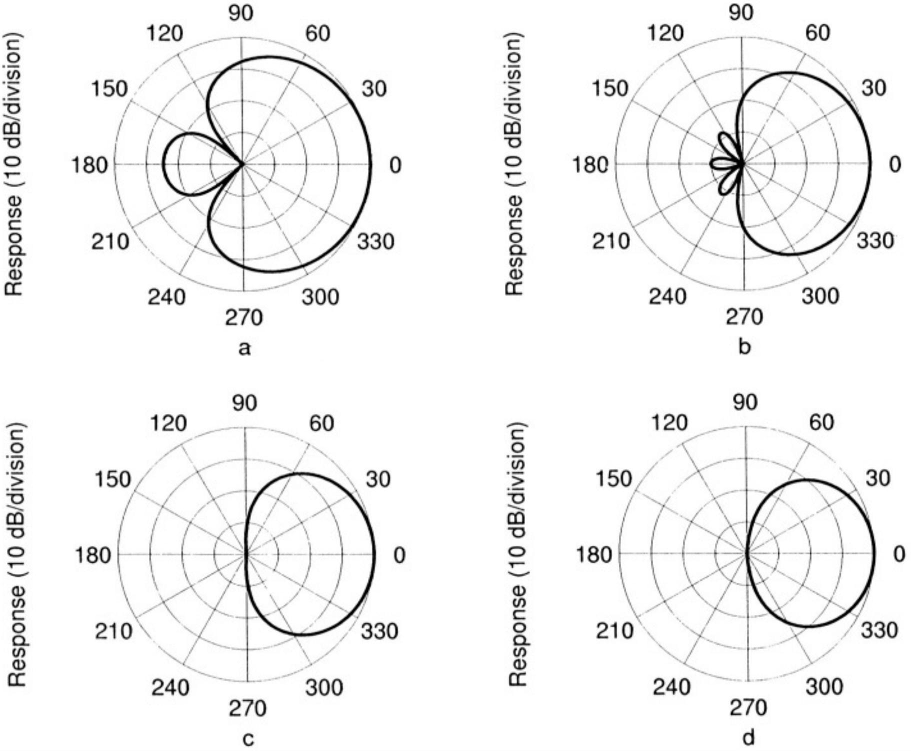


Figure 2.10 Directivity patterns for maximum front-to-back power ratio for differential arrays in a cylindrically isotropic noise field for (a) first, (b) second, (c) third, and (d) fourth-order.

4.4 MINIMUM PEAK DIRECTIONAL RESPONSE

Another approach that might be of interest, is to design differential arrays that have an absolute maximum sidelobe response. This specification would allow a designer to guarantee that the differential array response would not exceed a defined level over a given angular range, where suppression of acoustic signals is desired.

The first suggestion of an equi-sidelobe *differential* array design was in a "comment" publication by V. I. Korenbaum [9], who only discussed a restricted class of n^{th}-order microphones that have the following form:

$$E_{N_n}(\theta) = [\alpha_1 + (1 - \alpha_1)\cos\theta]\cos^{n-1}\theta. \tag{2.84}$$

The restricted class defined by (2.84) essentially assumes that an n^{th}-order differential microphone is the combination of an $(n-1)^{th}$-order dipole pattern and a general first-order pattern. The major reason for considering this restricted class is obvious; the algebra becomes very simple. Since we are dealing with systems of order less than or equal to three, we do not need to restrict ourselves to the class defined by (2.84).

A more general design of equi-sidelobe differential arrays can be obtained by using the standard Dolph-Chebyshev design techniques [6]. With this method, one can easily realize any order differential microphone. Roots of the Dolph-Chebyshev system are easily obtained. Knowledge of the roots simplifies the formulation of the canonic equations that describe n^{th} order microphones as products of first-order differential elements.

To begin an analysis, the Chebyshev polynomials are defined as

$$T_n(x) = \begin{cases} \cos(n \cos^{-1} x), & -1 < x < 1 \\ \cosh(n \cosh^{-1} x), & 1 \leq | x | \end{cases} . \tag{2.85}$$

Chebyshev polynomials of order n have n real roots for arguments between –1 and 1, and their value grows proportional to x^n for arguments with a magnitude greater than 1. Thus, designs of n^{th}-order Chebyshev arrays require a transformation of the variable x in (2.85). Using the substitution $x = b + a \cos\theta$ in (2.85), enables one to form a desired n^{th}-order directional response that follows the Chebyshev polynomial over any range. At $\theta = 0°$, $x = x_o \equiv a + b$, and the value of the Chebyshev polynomial is $T_n(x_o) \geq 1$. Setting this value to the desired mainlobe to sidelobe ratio L, we have

$$L = T_n(x_o) = \cosh(n \cosh^{-1} x_o), \tag{2.86}$$

or equivalently,

$$x_o = a + b = \cosh\left(\frac{1}{n} \cosh^{-1} L\right). \tag{2.87}$$

The sidelobe at $\theta = 180°$ corresponds to $x = b - a = -1$. Therefore

$$\begin{aligned} a &= \frac{x_o + 1}{2}, \\ b &= \frac{x_o - 1}{2}. \end{aligned} \tag{2.88}$$

Since the zeros of the Chebyshev polynomial are readily calculable, the null locations are easily found. From the definition of the Chebyshev polynomial given in (2.85), the zeros occur at

$$x_m = \cos\left[\frac{(2m-1)\pi}{2n}\right], \quad m = 1, \ldots, n. \tag{2.89}$$

The nulls are therefore at angles

$$\theta_m = \cos^{-1}\left(\frac{2x_m - x_o + 1}{x_o + 1}\right). \tag{2.90}$$

4.5 BEAMWIDTH

Another useful measure of the array performance is the beamwidth. The beamwidth can be defined in many ways. It can refer to the angle enclosed between the zeros of a directional response or the 3 dB points. We will use the 3 dB beamwidth definition in this chapter. For a first-order microphone response (2.17), the 3 dB beamwidth is simply

$$\theta_{B_1} = 2\cos^{-1}\left(\frac{-2a_0 + \sqrt{2}(a_0 + a_1)}{2a_1}\right). \tag{2.91}$$

For a second-order array, the algebra is somewhat more difficult but still straightforward. The result from (2.28) is

$$\theta_{B_2} = 2\cos^{-1}\left(\frac{-a_1 + \sqrt{a_1^2 + 2\sqrt{2}[a_2^2 + a_1a_2 + (1-\sqrt{2})a_0a_2]}}{2a_2}\right), \tag{2.92}$$

where it is assumed that $a_2 \neq 0$. If $a_2 = 0$, the microphone degenerates into a first order array and the beamwidth can be calculated by (2.91).

Similarly, the beamwidth for a third-order array can be found although the algebraic form is extremely lengthy and is therefore not included here.

5. DESIGN EXAMPLES

For differential microphones with interelement spacing much less than the acoustic wavelength, the maximum directivity index is attained when all of the microphones are collinear. For this case, the maximum directivity index is $20\log_{10}(n+1)$ where n is the order of the microphone [13]. For first, second, and third-order microphones, the maximum directivity indices are, 6.0, 9.5, and 12.0 dB respectively. Derivations of some design examples for $n \leq 3$ are given in the following sections.

As indicated in (2.25), there are an infinite number of possible designs for n^{th}-order differential arrays. Presently, the most common first-order microphones are: dipole, cardioid, hypercardioid, and supercardioid. The extension to higher orders is straightforward and is developed in later sections. Most of the arrays that are described in this chapter have directional characteristics that are optimal in some way; namely, the arrays are optimal with respect to one of the performance measures previously discussed: directivity index, front-to-back ratio, sidelobe threshold, and beamwidth. A summary of the results for first, second, and third-order microphones is given in Table 2.6 at the end of this section.

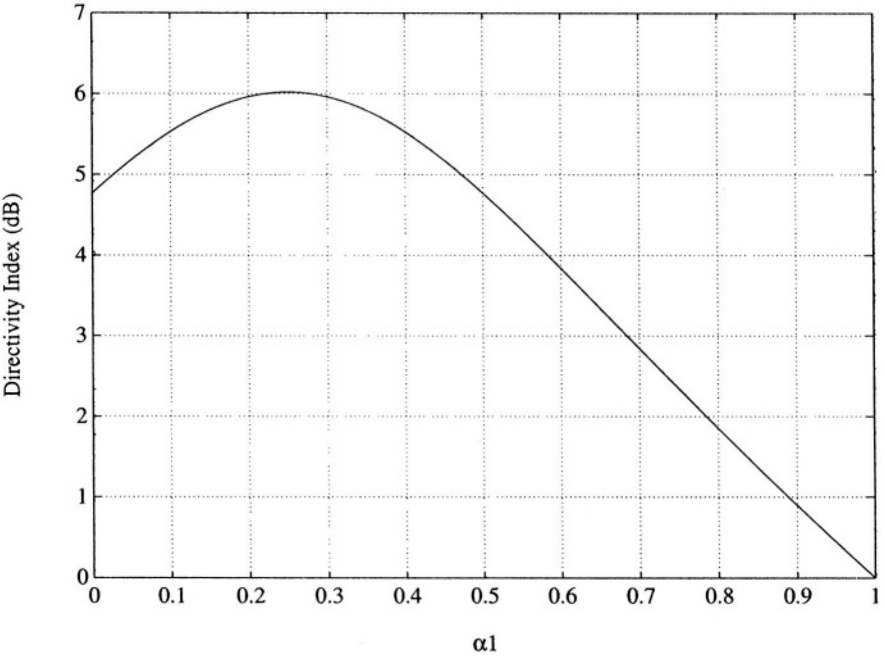


Figure 2.11 Directivity index of first-order microphone versus the first-order differential parameter α_1.

5.1 FIRST-ORDER DESIGNS

Before actual first-order differential designs are discussed it is instructive to first examine the effects of the parameter α_1 on the directivity index *DI*, the front-to-back ratio *F*, and the beamwidth of the microphone. For first-order arrays $\alpha_1 = a_0$ and $a_1 = 1 - \alpha_1$. Figure 2.11 shows the directivity index of a first-order system for values of α_1 between 0 and 1. The first-order differential microphone that corresponds to the maximum in Fig. 2.11 is given the name *hypercardioid.* When $\alpha_1 = 0$, the first-order differential system is a dipole. At $\alpha_1 = 1$, the microphone is an omnidirectional microphone with 0 dB directivity index. Figure 2.12 shows the dependence of the front-to-back ratio *F* on α_1. The maximum *F* value corresponds to the *supercardioid* design. Figure 2.13 shows the 3 dB beamwidth of the first-order differential microphone as a function of α_1.

When $\alpha_1 \approx 0.7$, the 3 dB down point is approximately at 180°. Higher values of α_1 correspond to designs that are increasingly omnidirectional and are sometimes referred to as subcardioid in the literature. Figure 2.13 indicates

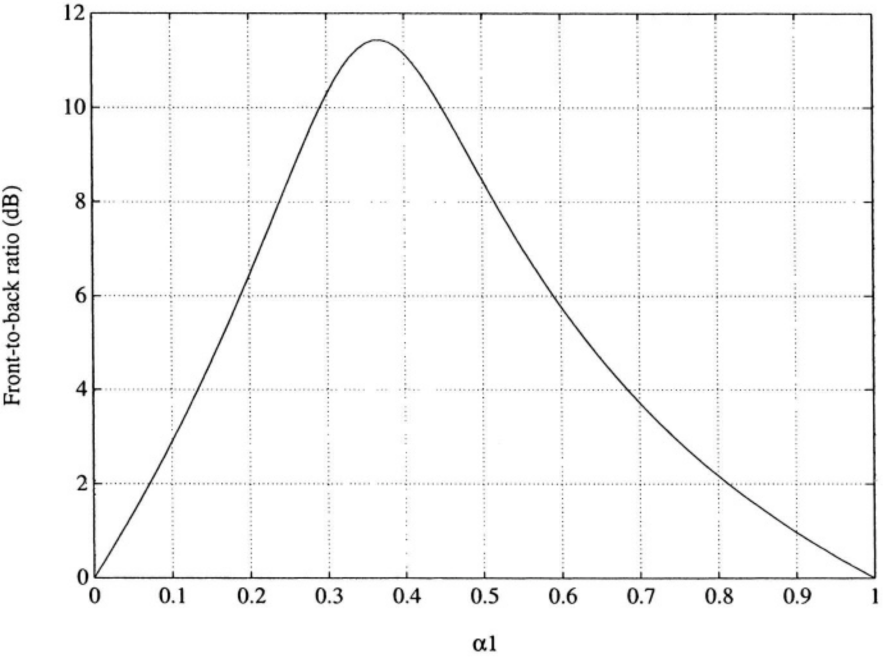


Figure 2.12 Front-to-back ratio of first-order microphone versus the first-order differential parameter α_1.

that the first-order differential microphone with the smallest beamwidth is the dipole microphone with a 3 dB beamwidth of 90°.

5.1.1 Dipole. From Euler's equation, it is evident that the dipole microphone is simply related to an acoustic particle-velocity microphone. The construction was described earlier; the dipole is normally realized as a diaphragm whose front and rear sides are directly exposed to the sound field. In (2.17), the dipole microphone corresponds the simple case where $a_0 = 0, a_1 = 1$,

$$E_{D_1}(\theta) = \cos\theta. \tag{2.93}$$

In Fig. 2.14(a), a polar plot of the magnitude of (2.93) shows the classic cosine pattern for the microphone.

A first-order dipole microphone directivity index is 4.8 dB and has a 3 dB beamwidth of 90°. A single null in the response is at $\theta = 90°$. One potential problem, however, is that it is bidirectional; in other words, the pattern is symmetric about the axis tangential to the diaphragm or normal to the axis of two

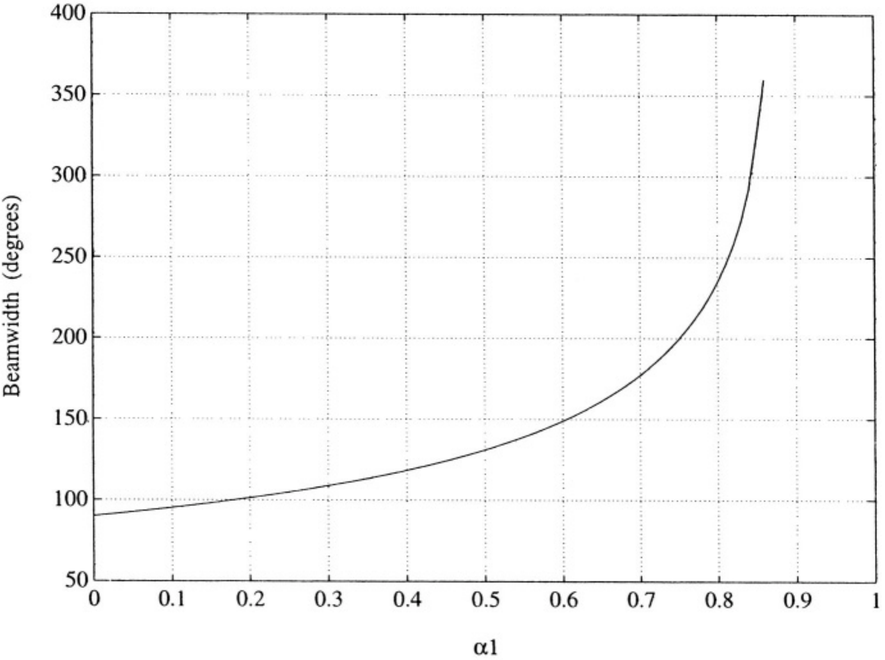


Figure 2.13 3 dB beamwidth of first-order microphone versus the first-order differential parameter α_1.

subtracted zero-order microphones that form a dipole. From pattern symmetry it is evident that the front-to-back ratio is 0 dB.

5.1.2 Cardioid. As shown earlier, all first-order patterns correspond to the "limaçon of Pascal" algebraic form. The special case of $\alpha_1 = 1/2$ is the cardioid pattern. The pattern is described by

$$E_{C_1}(\theta) = \frac{1 + \cos\theta}{2} \tag{2.94}$$

which is plotted in Fig. 2.14(b). Although the cardioid microphone is not optimal in directional gain or front-to-back ratio, it is the most commonly manufactured differential microphone. The cardioid directivity index is 4.8 dB, the same as that of the dipole microphone and the 3 dB beamwidth is 131°. A null in the response is located at $\theta = 180°$ and the front-to-back ratio is 8.5 dB.

5.1.3 Hypercardioid. The hypercardioid microphone has the distinction of having the highest directivity index of any first-order microphone. From the

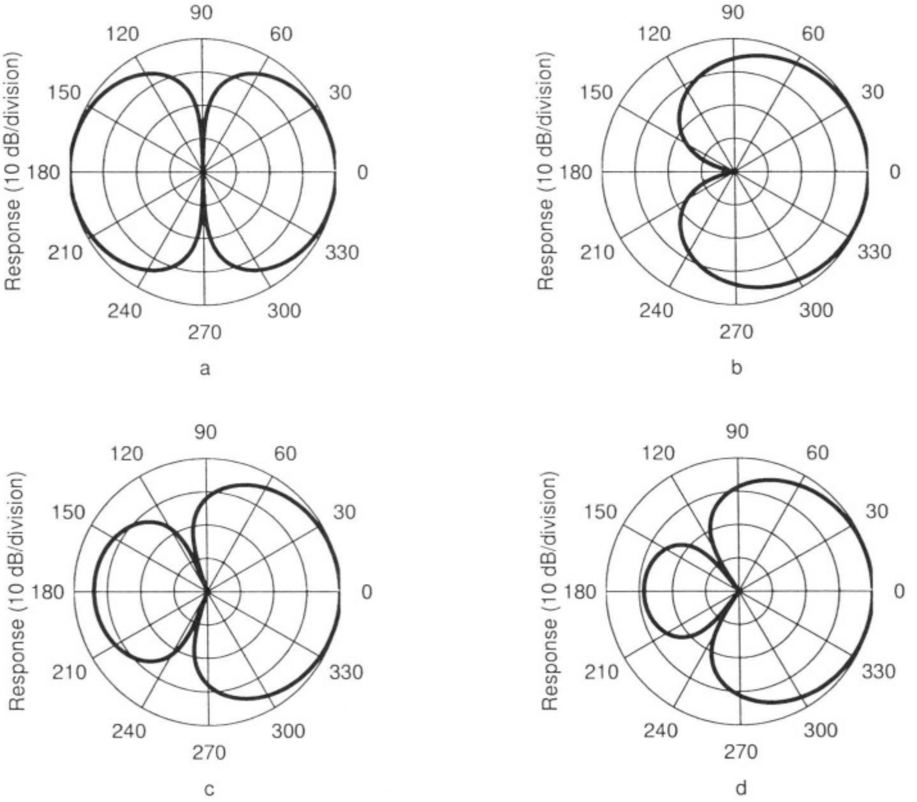


Figure 2.14 **Various first-order directional responses, (a) dipole, (b) cardioid, (c) hypercardioid, (d) supercardioid.**

previous section that discussed optimal arrays in a spherically isotropic field, the first-order hypercardioid response can be written as

$$E_{HC_1}(\theta) = \frac{1 + 3\cos\theta}{4}. \qquad (2.95)$$

Figure 2.14(c) is a polar plot of the absolute value of (2.95). The 3 dB beamwidth is equal to 105° and the null is at 109°. The directivity index is 6 dB or $10 \log_{10}(4)$, the maximum directivity index for a first-order system and the front-to-back ratio is 8.5 dB.

5.1.4 Supercardioid. The name *supercardioid* is commonly used for a first-order differential design which maximizes the front-to-back received power. Apparently, the first reference to the supercardioid design appears in a 1941 paper by Marshall and Harry [11]. A supercardioid is of interest since of all first-order designs it has the highest front-to-back power rejection for isotropic noise. A supercardioid response can be written as

$$E_{SC_1}(\theta) = \frac{\sqrt{3} - 1 + (3 - \sqrt{3}) \cos \theta}{2}. \tag{2.96}$$

Figure 2.14(d) is a plot of the magnitude of (2.96). The directivity index for the supercardioid is 5.7 dB with a 3 dB beamwidth of 115°. A null in the response is located at 125°. The front-to-back ratio is 11.4 dB.

5.2 SECOND-ORDER DESIGNS

As with first-order systems, there are an unlimited number of second-order array designs. Since second-order microphones are not readily available on the market today, there are no "common" configurations. Two designs that have been suggested are the second-order cardioid and the second-order hypercardioid [12, 15]. Another group of proposed differential microphones is a restricted class of equi-sidelobe designs for arbitrary order n [9]. This section presents some of these designs as well as non-restricted equi-sidelobe designs and also a variety of second-order differential array designs based on common first-order microphones. The general second-order form as given in (2.28) has three parameters, a_0, a_1, and a_2. Equivalently, the second-order differential is the product of two first-order differential forms as shown in (2.29).

It is informative to plot the directivity index and front-to-back ratio as a function of the canonical values of α_1 and α_2. One can then easily visualize how these measures change for different second-order designs. Figures 2.15 and 2.16 depict the dependence of *DI* and *F* on the two independent parameters α_1 and α_2 from (2.29). Both figures are plotted for values of α_1 and α_2 between −1 and +1. Figure 2.15 shows the *DI* maximum value of 9.5 dB and the interval between the contours is 0.5 dB. The two peaks in the plot represent the same maximum and only the order of the product of first-order sections used to represent the second-order response has changed. Figure 2.16 shows the result for the front-to-back ratio which has a maximum value of 24.0 dB; the contours are in 1 dB steps.

5.2.1 Second-Order Dipole.

By pattern multiplication, the second-order dipole directional response is the product of two first-order dipoles which have $\cos\theta$ response patterns given by

$$E_{D_2}(\theta) = \cos^2 \theta. \tag{2.97}$$

Figure 2.17(a) shows the polar magnitude response for this array. The directivity index is 7.0 dB, and by symmetry the front-to-back ratio is 0 dB. The 3 dB beamwidth is 65°.

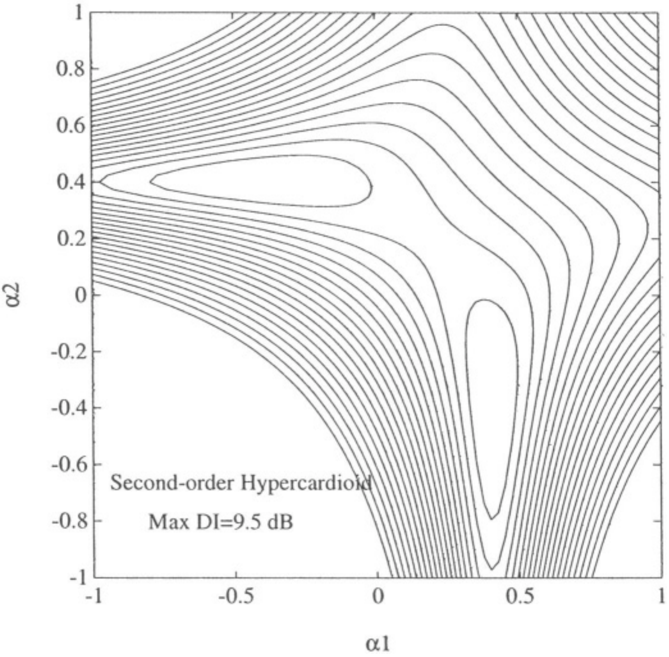


Figure 2.15 Contour plot of the directivity index DI in dB for second-order array versus α_1 and α_2. The contours are in 0.5 dB intervals.

5.2.2 Second-Order Cardioid. In general the term second-order cardioid implies that either first-order term in the second-order expression given in (2.29) can be a cardioid. A simple second-order cardioid corresponds to the Case when both first-order terms are of the cardioid form

$$E_{C_2}(\theta) = \frac{(1+\cos\theta)^2}{4}. \tag{2.98}$$

For this special case $\alpha_1 = \alpha_2 = 0.5$ and the directivity index is 7.0 dB and both nulls fall at $\theta = 180°$. The front-to-back ratio is 14.9 dB.

A more general form for a second-order array can be written as the product of a first-order array with that of a first-order cardioid. An equation for this second-order cardioid is,

$$E_{FC_2}(\alpha_1,\theta) = \frac{[\alpha_1 + (1-\alpha_1)\cos\theta][1+\cos\theta]}{2}. \tag{2.99}$$

5.2.3 Second-Order Hypercardioid. The second-order hypercardioid has the highest directivity index of a second-order system; its directivity in-

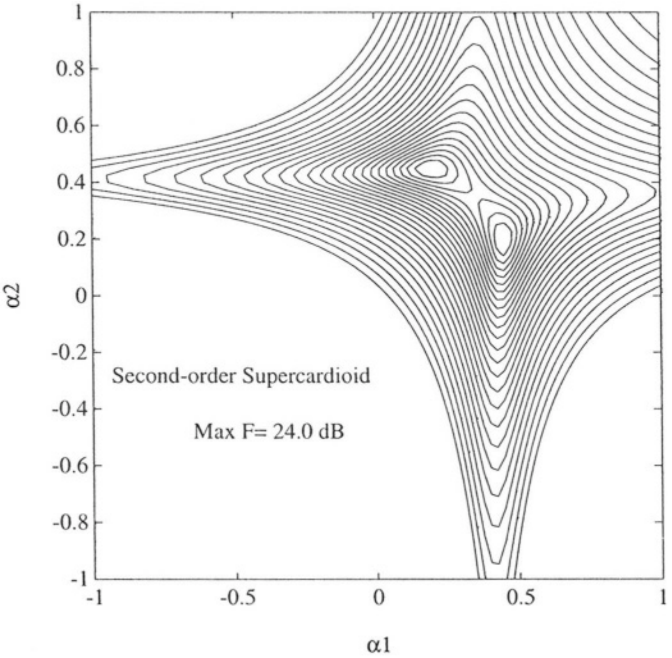


Figure 2.16 Contour plot of the front-to-back ratio in dB for second-order arrays versus α_1 and α_2. The contours are in 1 dB increments.

dex is 9.5 dB. A derivation of the directivity pattern and the parameters that determine the second-order hypercardioid are contained in Section 3. The results are:

$$\alpha_1 = \pm\frac{1}{\sqrt{6}} \approx \pm 0.41, \tag{2.100}$$

$$\alpha_2 = \mp\frac{1}{\sqrt{6}} \approx \mp 0.41. \tag{2.101}$$

These values correspond to the peaks in Fig. 2.15. Null locations for the second-order hypercardioid are at 73° and 134°. The front-to-back ratio is 8.5 dB, which is the same for the first-order differential cardioid and first-order differential hypercardioid. A polar response is shown in Fig. 2.17(c).

5.2.4 Second-Order Supercardioid. The term second-order supercardioid designates an optimal design for the second-order differential microphone with respect to the front-to-back received power ratio. A derivation for the supercardioid microphone was also given in Section 3 and the results (repeated

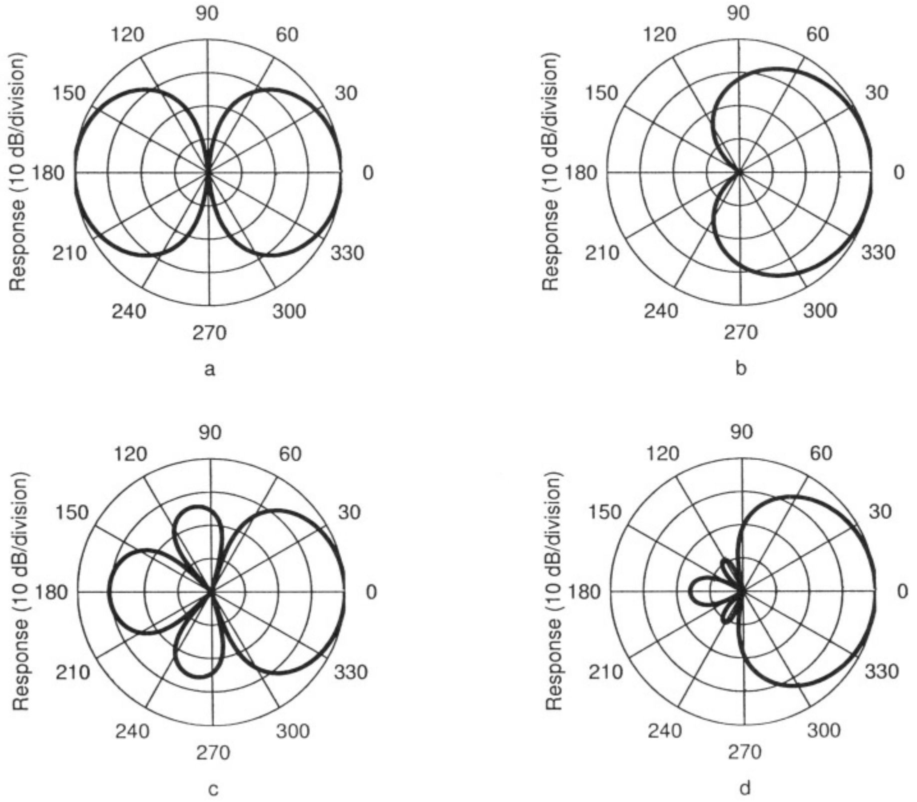


Figure 2.17 Various second-order directional responses, (a) dipole, (b) cardioid, (c) hypercardioid, (d) supercardioid.

here) are:

$$\alpha_1 = \frac{\sqrt{7} - 2 \pm \sqrt{8 - 3\sqrt{7}}}{2} \approx 0.45,\ 0.20, \tag{2.102}$$

$$\alpha_2 = \frac{\sqrt{7} - 2 \mp \sqrt{8 - 3\sqrt{7}}}{2} \approx 0.20,\ 0.45. \tag{2.103}$$

These values correspond to the peaks in Figure 2.16. Figure 2.17(d) is a plot of the magnitude of the directional response. The directivity index is equal to 8.3 dB, the nulls are located at 104° and 144°, while front-to-back ratio is 24.0 dB.

5.2.5 Equi-Sidelobe Second-Order Differential. Since a second-order differential microphone has two zeros in its response it is possible to design a second-order microphone such that the two lobes defined by these zeros are at the same level. Figure 2.18(a) shows the *only* second-order equi-sidelobe design

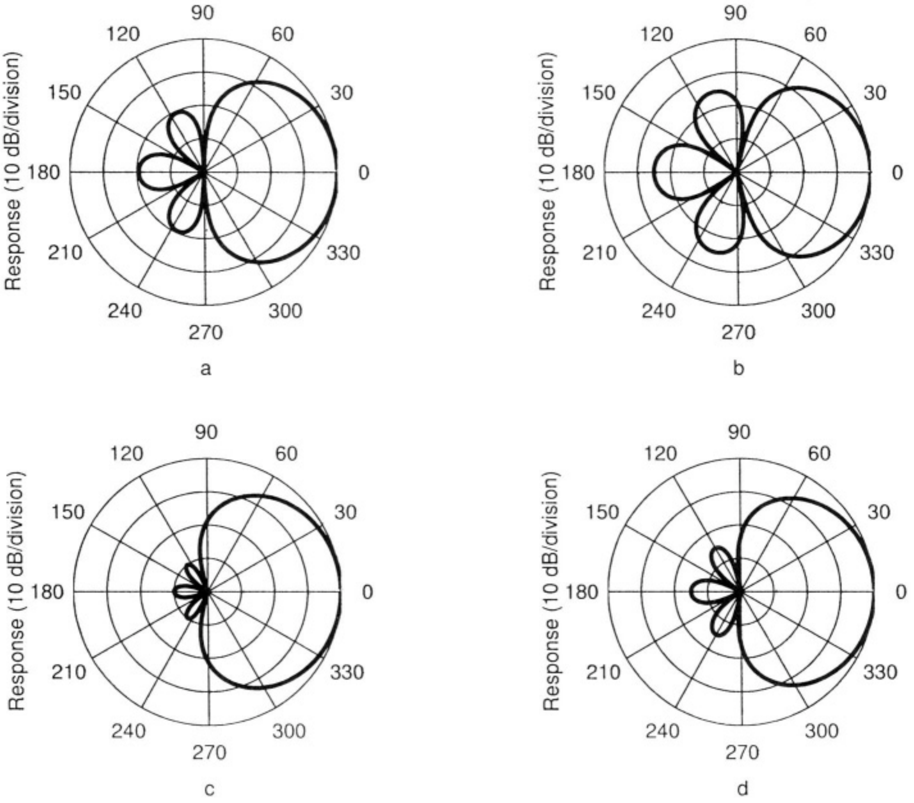


Figure 2.18 Various second-order equi-sidelobe designs, (a) Korenbaum design, (b) −15 dB sidelobes, (c) −30 dB sidelobes, (d) minimum rear half-plane peak response.

possible using the form of (2.84). The directivity index of this *Korenbaum* second-order differential array is 8.9 dB. The beamwidth is 76° and the front-to-back ratio is 17.6 dB.

We begin our analysis of second-order Chebyshev differential arrays begins by comparing terms of the Chebyshev polynomial and the second-order array response function. The Chebyshev polynomial of order 2 is

$$\begin{aligned} T_2(x) &= 2x^2 - 1 \\ &= 2b^2 - 1 + 4ab\cos\theta + 2a^2\cos^2\theta. \end{aligned} \tag{2.104}$$

Comparing like terms of (2.104) and (2.28) yields:

$$\begin{aligned} a_0 &= \frac{2b^2-1}{L}, \\ a_1 &= \frac{4ab}{L}, \\ a_2 &= \frac{2a^2}{L}, \end{aligned} \tag{2.105}$$

where L is again the sidelobe threshold level.

By substituting the results of (2.88) into (2.105), we can determine the necessary coefficients for the desired equi-sidelobe second-order differential microphone:

$$\begin{aligned} a_0 &= \frac{x_o^2-2x_o-1}{2S}, \\ a_1 &= \frac{x_o^2-1}{S}, \\ a_2 &= \frac{(x_o+1)^2}{2S}, \end{aligned} \tag{2.106}$$

where

$$x_o = \cosh\left(\frac{1}{2}\cosh^{-1} S\right). \tag{2.107}$$

Thus for the second-order differential microphone,

$$\theta_{1,2} = \cos^{-1}\left(\frac{1-x_o \pm \sqrt{2}}{1+x_o}\right). \tag{2.108}$$

The null locations given in (2.108) can be used along with (2.32) and (2.33) to determine the canonic first-order differential parameters α_1 and α_2. Figures 2.18(b) and 2.18(c) show the resulting second-order designs for –15 dB and –30 dB sidelobes respectively. The directivity indices for the two designs are respectively 9.4 dB and 8.1 dB. Null locations for the –15 dB sidelobes design are at 78° and 142°. By allowing a higher sidelobe level than the Korenbaum design (for $x_o = 1+\sqrt{2}$, with $\theta_1 = 90°$), a higher directivity index can be achieved. In fact, the directivity index monotonically increases until the sidelobe levels exceed –13 dB; at this point the directivity index reaches its maximum at 9.5 dB, almost the maximum directivity index for a second-order differential microphone. For sidelobe levels less than –20.6 dB (Korenbaum design), both nulls are in the rear half-plane of the second-order microphone; null locations for the –30 dB sidelobes design are at 109 and 152 degrees. Equi-sidelobe second-order directional patterns always contain a peak lobe at $\theta = 180°$.

One interesting design that arises from the preceding development is a second-order differential microphone that minimizes the *peak* rear half-space response. This design corresponds to the case where the front-lobe response level at $\theta = 90°$ is equal to the equi-sidelobe level (for $x_o = 3$). Figure 2.18(d) is a directional plot of this realization. The canonic first-order differential parameters for this equi-sidelobe design are:

$$\begin{aligned} \alpha_1 &= \frac{5 \pm 2\sqrt{2}}{17}, \\ \alpha_2 &= \frac{5 \mp 2\sqrt{2}}{17}. \end{aligned} \tag{2.109}$$

This design has a directivity index of 8.5 dB and nulls located at 149° and 98°, The front-to-back ratio is 22.4 dB.

Two other design possibilities can be obtained by determining the equi-sidelobe second-order design that maximizes either the directivity index *DI* or the front-to-back ratio *F*. Figure 2.19 is a plot of the directivity and front-to-back indices as a function of sidelobe level. As mentioned earlier, a –13 dB sidelobe level maximizes the directivity index at 9.5 dB. A sidelobe level of –27.5 dB maximizes the front-to-back ratio. Plots of these two designs are shown in Fig. 2.20. Of course, an arbitrary combination of *DI* and *F* could also be maximized for some given optimality criterion if desired.

5.2.6 Maximum Second-Order Differential *DI* and *F* Using Common First-Order Differential Microphones. Another approach to the design of second-order differential microphones involves the combination of the outputs of two first-order differential microphones. Specifically, the combination is a subtraction of the first-order differential outputs after one is passed through a delay element. If the first-order differential microphone can be designed to have any desired canonic parameter α_1, then *any* second-order differential array can be designed. More commonly a designer will have to work with off-the-shelf first-order differential microphones, such as standard first-order designs. If the second-order design is constrained to standard first-order differential microphones, then it is not possible to reach the maximum directivity index and front-to-back ratio. The following section discusses how to implement an optimal design with respect to directivity index and front-to-back power rejection when using standard first-order microphone elements.

Given a first-order differential microphone with the directivity function,

$$E_{N_1}(\alpha_2, \theta) = \alpha_2 + (1 - \alpha_2)\cos\theta, \tag{2.110}$$

where α_2 is a constant, it is of interest to know how to combine two of these microphones so that the directivity index is maximized. A maximum can be found by multiplying (2.110) by a general first-order response, integrating the

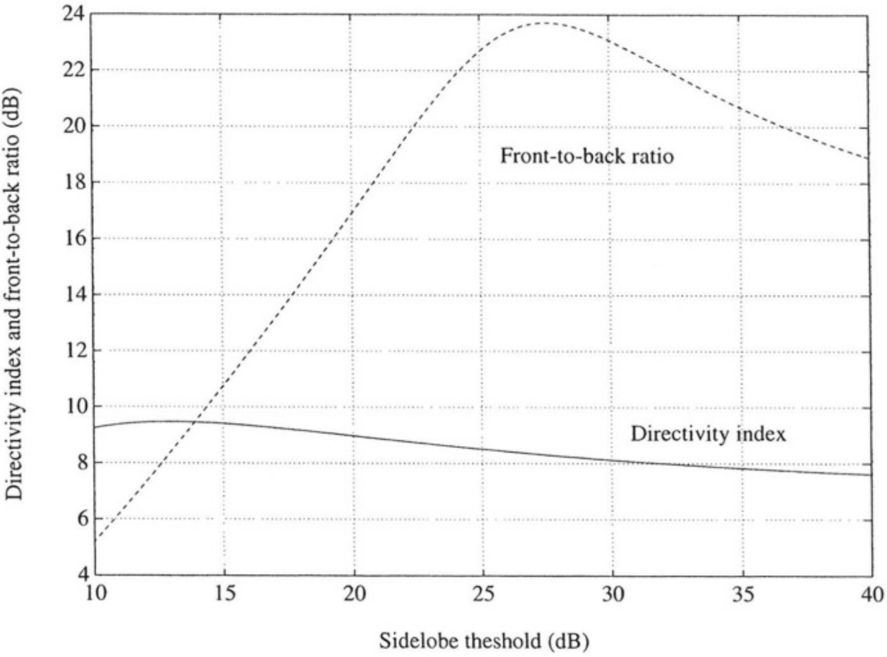


Figure 2.19 Directivity index (solid) and front-to-back ratio (dotted) for equi-sidelobe second-order array designs versus sidelobe level.

square of this product from $\theta = 0$ to π, taking derivative with respect to α_1, and setting the resulting derivative to zero. The result is

$$\alpha_1 = \frac{3}{8} - \frac{5\alpha_2}{8(2 - 9\alpha_2 + 12\alpha_2^2)}. \tag{2.111}$$

A plot of the directivity index for $0 \leq \alpha_2 \leq 1$ is shown in Fig. 2.21. A maximum value of 9.5 dB occurs when $-\alpha_1 = \alpha_2 \approx 0.41$.

A similar calculation for the maximum front-to-back power response yields a rather long expression:

$$\alpha_1 = \frac{\beta - 8\alpha_2^3 - 12\alpha_2^2 + 13\alpha_2 - 3}{24\alpha_2^4 - 6\alpha_2 + 2}, \tag{2.112}$$

where

$$\beta = \sqrt{8\alpha_2^4 + 8\alpha_2^3 + 8\alpha_2^2 - 12\alpha_2 + 3}\sqrt{12\alpha_2^4 + 6\alpha_2^3 + 17\alpha_2^2 - 20\alpha_2 + 5}\,. \tag{2.113}$$

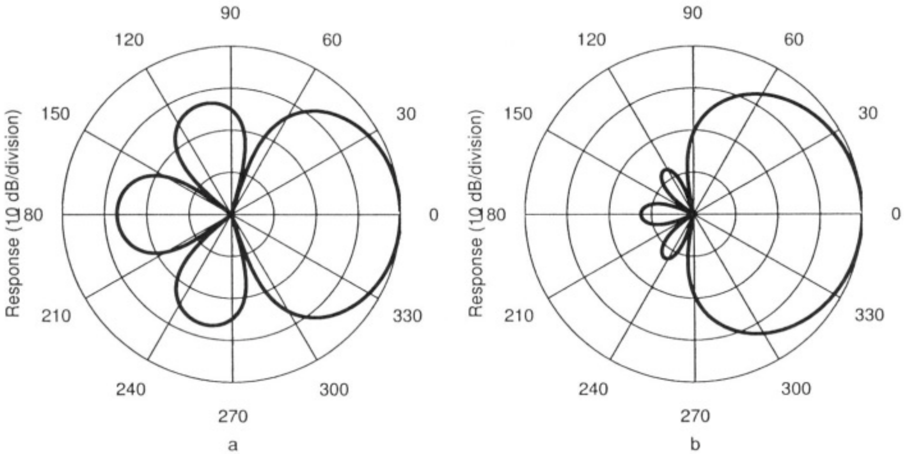


Figure 2.20 Directional responses for equi-sidelobe second-order differential arrays for, (a) maximum directivity index, and, (b) maximum front-to-back ratio.

A plot of the front-to-back ratio for $0 \leq \alpha_2 \leq 1$ is shown in Fig. 2.22.

A maximum value of 24.0 dB occurs when $\alpha_2 \approx 0.45$ and $\alpha_1 \approx 0.20$, which are the values of the second-order supercardioid. By symmetry, the values of α_1 and α_2 can obviously be interchanged. The double peak at F = 24.0 dB in Figure 2.22 is a direct result of this symmetry.

5.3 THIRD-ORDER DESIGNS

Very little can be found in the open literature on the construction and design of third-order differential microphones. The earliest paper in which an actual device was designed and constructed was by B. R. Beavers and R. Brown in 1970 [2]. The lack of any papers on third-order arrays is not surprising given both the extreme precision that is necessary to realize these arrays and the serious signal-to-noise problems (these problems are discussed in more detail in a later section). However, recent advances in low noise microphones and electronics support the feasibility of third-order microphone construction. With this in mind, the following section describes several possible design implementations.

5.3.1 Third-Order Dipole. By pattern multiplication, the third-order dipole directional response is given by

$$E_{D_3}(\theta) = \cos^3 \theta. \qquad (2.114)$$

Figure 2.23(a) shows the magnitude response for this array. The directivity

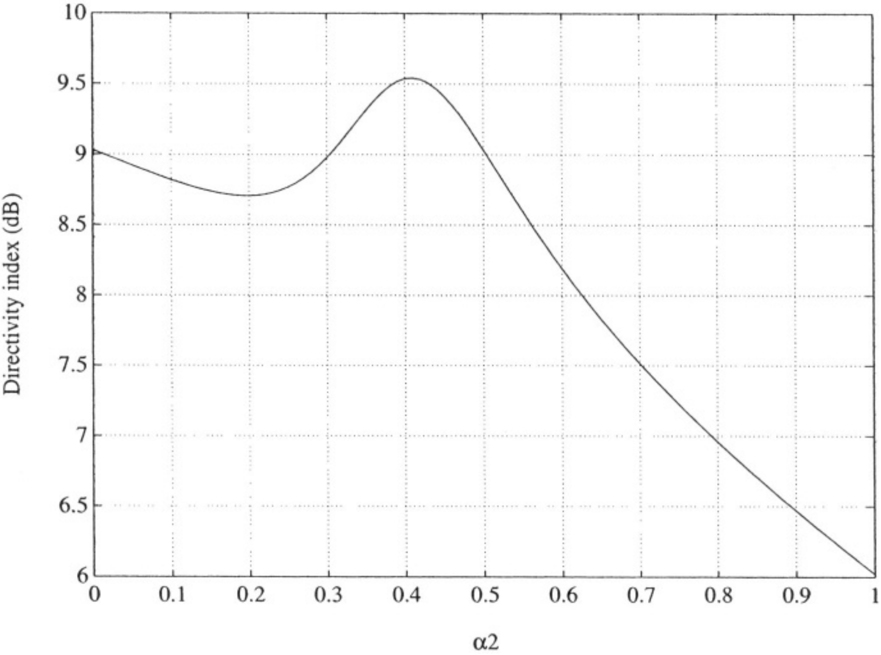


Figure 2.21 Maximum second-order differential directivity index DI for first-order differential microphones defined by (2.110).

index is 8.5 dB, while the front-to-back ratio is 0 dB and the 3 dB beamwidth is 54°.

5.3.2 Third-Order Cardioid. The terminology of cardioids is ambiguous for second-order arrays. For the third-order, this ambiguity is even more pervasive. Nevertheless an obvious array possibility is to form a cardioid by using the pattern multiplication of three first-order differential cardioids, then

$$E_{c_3} = \frac{(1+\cos\theta)^3}{8}. \tag{2.115}$$

Figure 2.23(b) shows the directional response for this array. The three nulls all fall at 180°. The directivity index is 8.5 dB and the front-to-back ratio is 21.0 dB.

5.3.3 Third-Order Hypercardioid. A derivation for the third-order differential hypercardioid was given in Section 4.2. The results for the coefficients

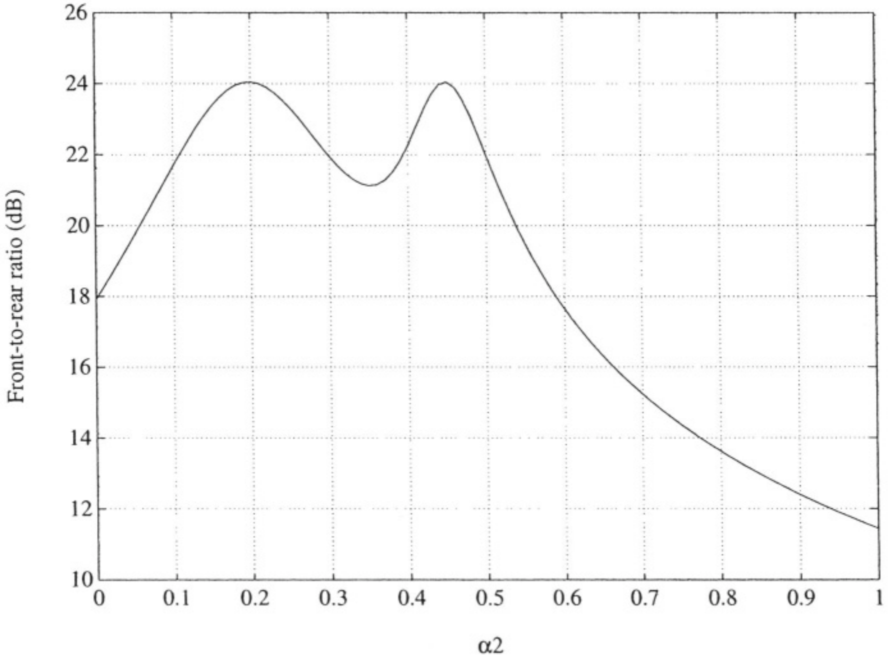


Figure 2.22 Maximum second-order differential front-to-back ratio for first-order differential microphones defined by (2.110).

in (2.34) are:

$$\begin{aligned} a_0 &= -3/32, \\ a_1 &= -15/32, \\ a_2 &= 15/32, \\ a_3 &= 35/32. \end{aligned} \tag{2.116}$$

After solving for the roots of (2.34) with the coefficients given in (2.116), the coefficients of the canonic representation are:

$$\begin{aligned} \alpha_1 &= 1/2\left[\sqrt{5}\cos(\phi/3) - 1/2\right] \approx 0.45, \\ \alpha_2 &= 1/2\left[\sqrt{5}\cos(\phi/3 + 2\pi/3) - 1/2\right] \approx 0.15, \\ \alpha_3 &= 1/2\left[\sqrt{5}\cos(\phi/3 + 4\pi/3) - 1/2\right] \approx -1.35, \end{aligned} \tag{2.117}$$

where

$$\phi = \arccos\left(-2/\sqrt{5}\right). \tag{2.118}$$

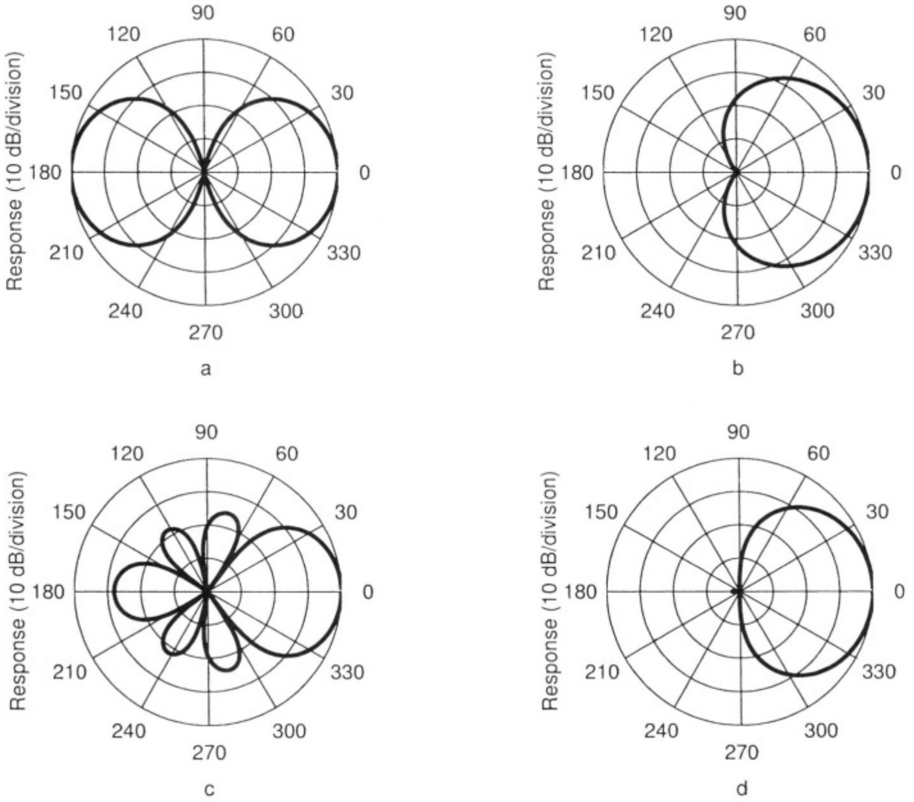


Figure 2.23 **Various third-order directional responses, (a) dipole, (b) cardioid, (c) hypercardioid, (d) supercardioid.**

Figure 2.23(c) shows the directional response of the third-order differential hypercardioid. The directivity index of the third-order differential hypercardioid is the maximum for all third-order designs: 12.0 dB. The front-to-back ratio is 11.2 dB and the three nulls are located at $\theta = 55°$, 100°, and 145°.

5.3.4 Third-Order Supercardioid. The third-order supercardioid is the third-order differential array with the maximum front-to-back power ratio. The derivation of this array was given in Section 4.3. The requisite coefficients a_i

are:

$$\begin{aligned}
a_0 &= \frac{\sqrt{2}\sqrt{21-\sqrt{21}}-\sqrt{21}-1}{8} \approx 0.018, \\
a_1 &= \frac{21+9\sqrt{21}-\sqrt{2}(6+\sqrt{21})\sqrt{21-\sqrt{21}}}{8} \approx 0.200, \\
a_2 &= \frac{3[\sqrt{2}(4+\sqrt{21})\sqrt{21-\sqrt{21}}-25-5\sqrt{21}]}{8} \approx 0.475, \\
a_3 &= \frac{63+7\sqrt{21}-\sqrt{2}(7+2\sqrt{21})\sqrt{21-\sqrt{21}}}{8} \approx 0.306. \qquad (2.119)
\end{aligned}$$

Figure 2.23(d) shows a directivity plot of the resulting supercardioid microphone. The directivity index is 9.9 dB and the front-to-back ratio is 37.7 dB. The nulls are located at 97°, 122°, and 153°. This third-order supercardioid has almost no sensitivity to the rear half-plane. For situations where the user desires information from only one half-plane, the third-order supercardioid microphone performs optimally. Finding the roots of (2.34) with the coefficients given by (2.119) yields the parameters of the canonic expression given in (2.35) as

$$\begin{aligned}
\alpha_1 &\approx .113, \\
\alpha_2 &\approx .473, \\
\alpha_3 &\approx .346. \qquad (2.120)
\end{aligned}$$

5.3.5 Equi-Sidelobe Third-Order Differential. Finally, the design of equi-sidelobe third-order differential arrays is explored. Like the design of equi-sidelobe second-order differential microphones, a third-order equi-sidelobe array relies on the use of Chebyshev polynomials and the Dolph-Chebyshev antenna synthesis technique. The basic technique was discussed earlier in Section 2.3. For a third-order microphone, the Chebyshev polynomial is

$$T_3(x) = 4x^3 - 3x. \qquad (2.121)$$

Using the transformation $x = b + a \cos \theta$ and comparing terms with (2.34) leads to

$$\begin{aligned}
a_0 &= \frac{b(4b^2-3)}{L}, \\
a_1 &= \frac{3a(4b^2-1)}{L}, \\
a_2 &= \frac{12a^2b}{L}, \\
a_3 &= \frac{4a^3}{L}. \qquad (2.122)
\end{aligned}$$

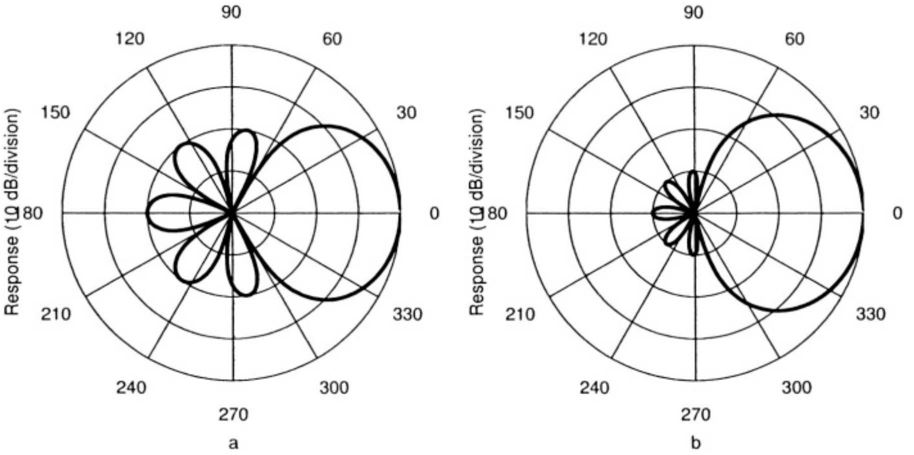


Figure 2.24 Equi-sidelobe third-order differential microphone for (a) −20 dB and (b) −30 dB sidelobes.

Combining (2.86), (2.87), (2.88), and (2.122) yields the coefficients for the equi-sidelobe third-order differential. These results are:

$$\begin{aligned} a_0 &= \frac{x_o^3 - 3x_o^2 + 2}{2L}, \\ a_1 &= \frac{3x_o(x_o + 1)(x_o - 2)}{2L}, \\ a_2 &= \frac{3(x_o - 1)(x_o + 1)^2}{2L}, \\ a_3 &= \frac{(x_o + 1)^3}{2L}. \end{aligned} \tag{2.123}$$

Figures 2.24(a) and 2.24(b) show the resulting patterns for –20 dB and –30 dB sidelobe levels. From (2.90), the nulls for the –20 dB sidelobe levels are at 62°, 102°, and 153°. The nulls for the –30 dB sidelobe case are at 79°, 111°, and 156°. The directivity indices are 11.8 dB and 10.8 dB, respectively. The front-to-back ratio for the –20 dB design is 14.8 dB and 25.2 dB for the –30 dB design.

Next, the directivity index and the front-to-back ratio for the equi-sidelobe third-order array as a function of sidelobe level is examined. Figure 2.25 shows these two quantities for equi-sidelobe levels from –10 dB to – 60 dB. The directivity index reaches its maximum at 12.0 dB for a sidelobe level of approximately –16 dB. The –16 dB equi-sidelobe design plotted in Fig. 2.26(a), is therefore close to the optimal third-order differential hypercardioid of Fig. 2.23(c). The front-to-back ratio reaches a maximum of 37.3 dB at a sidelobe level of

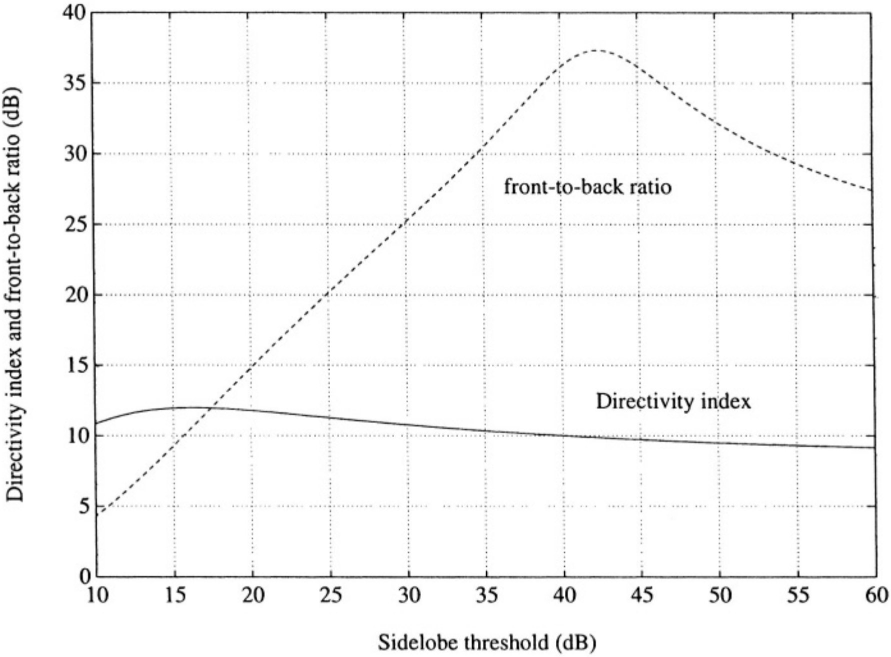


Figure 2.25 Directivity index and front-to-back ratio for equi-sidelobe third-order differential array designs versus sidelobe level.

– 42.5 dB; the response is plotted in Fig. 2.26(b). For sidelobe levels less than – 42.5 dB, the mainlobe moves into the rear half-plane. For sidelobe levels greater than – 42.5 dB, the zero locations move towards $\theta = 0°$ and as a result the beamwidth decreases.

5.4 HIGHER-ORDER DESIGNS

Due to sensitivity to electronic noise and microphone matching requirements, differential array designs higher that third-order are not practically realizable. These arrays will probably never be implemented on anything other than in a computer simulation. In fact, the design of higher-order supercardioid and hypercardioid differential arrays using the techniques discussed in Sections 4.2 and 4.3, can become computationally difficult on present computers.

Table 2.6 Table of first-order, second-order, and third-order differential microphone array designs for spherically isotropic acoustic fields.

microphone type	DI (dB)	F (dB)	Beamwidth	Null(s) (degs)
First-order designs				
dipole	4.8	0.0	90°	90
cardioid	4.8	8.5	131°	180
hypercardioid	6.0	8.5	105°	109
supercardioid	5.7	11.4	115°	125
Second-order designs				
dipole	7.0	0.0	65°	90
cardioid	7.0	14.9	94°	180
hypercardioid	9.5	8.5	66°	73, 134
supercardioid	8.3	24.0	80°	104, 144
−15 dB sidelobe	9.4	10.7	70°	78, 142
−30 dB sidelobe	8.1	18.5	84°	109, 152
min. rear peak	8.5	22.4	80°	98, 149
Third-order designs				
dipole	8.5	0.0	54°	90
cardioid	8.5	21.0	78°	180
hypercardioid	12.0	11.2	48°	55, 100, 145
supercardioid	9.9	37.7	66°	97, 122, 153
−20 dB sidelobe	11.8	14.8	52°	62, 102, 153
−30 dB sidelobe	10.8	25.2	60°	79, 111, 156

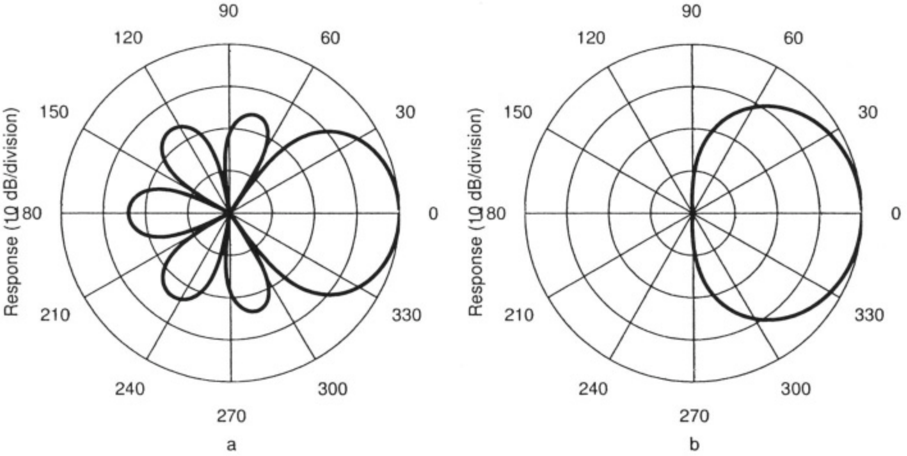


Figure 2.26 Directivity responses for equi-sidelobe third-order differential arrays for (a) maximum directivity index and (b) maximum front-to-back ratio.

6. SENSITIVITY TO MICROPHONE MISMATCH AND NOISE

There is a significant amount of literature on the sensitivity of superdirectional array design to interelement errors in position, amplitude, and phase [7, 10, 5]. Since the array designs discussed in this chapter have interelement spacings which are much less than the acoustic wavelength, differential arrays are indeed superdirectional arrays. Early work in superdirectional for *supergain* arrays involved over-steering a Dolph-Chebyshev array past endfire. When the effective interelement spacing becomes much less than the acoustic wavelength, the amplitude weighting of the elements oscillate between plus and minus, resulting in pattern differencing or differential operation. Curiously though, the papers in the field of superdirectional arrays never point out that at small spacings the array can be designed as a differential system as given by (2.25). A usual comment in the literature is that the design of superdirectional arrays requires amplitude weighting that is highly frequency dependent. For the application of the designs that we are discussing, namely differential systems where the wavelength is much larger than the array size, the amplitude weighting is constant with frequency as long as we do not consider the necessary time delay as part of the weighting coefficient. The only frequency correction necessary is the compensation of the output of the microphone for the ω^n high-pass characteristic of the n^{th} order system.

One quantity which characterizes the sensitivity of the array to random amplitude and position errors is the sensitivity function introduced by Gilbert and

Morgan [7]. The sensitivity function modified by adding a delay parameter τ is

$$K = \frac{\sum_{m=1}^{n} \mid b_m \mid^2}{\mid \sum_{m=1}^{n} b_m e^{-jk(r_m + c\tau_m)} \mid^2}, \qquad (2.124)$$

where r_m is the distance from the origin to microphone m, b_m are the amplitude shading coefficients of a linear array, and τ_m is the delay associated with microphone m. For the differential microphones discussed, the sensitivity function reduces to

$$K = \frac{n+1}{[\prod_{m=1}^{n} 2 \sin(k\,(d + c\tau_m)/2)]^2}, \qquad (2.125)$$

where d is the microphone spacing. For values of $kd \ll 1$, (2.125) can be further reduced to

$$K \approx \frac{n+1}{[\prod_{m=1}^{n} k(d + c\tau_m)]^2}. \qquad (2.126)$$

For the n^{th}-order dipole case, (2.126) reduces to

$$K_D \approx \frac{n+1}{(kd)^{2n}}. \qquad (2.127)$$

The ramifications of (2.127) are intuitively appealing. The equation merely indicates that the sensitivity to noise and array errors for n^{th}-orderdifferential arrays is inversely proportional to the frequency response of the array.

The response of an array to perturbations of amplitude, phase, and position can be expressed as a function of a common error term δ^2. The validity of combining these terms into one quantity hinges on the assumption that these errors are small compared to the desired values. The reader is referred to the article by Gilbert and Morgan for specific details [7].

The error perturbation power response pattern is dependent on the error term δ^2, the actual desired beam pattern, and the sensitivity factor K. The response is given by

$$E_{N_{np}}(\theta) = E_{N_n}(\theta) + K\,\delta^2. \qquad (2.128)$$

Typically δ is very small, and can be controlled by careful design. However, even with careful control, we can only hope for 1% tolerances in amplitude and position. Therefore, even under the best of circumstances there will be great difficulty in realizing a differential array if the value of K approaches or exceeds 10,000 (40 dB). A plot of the value of K for various first-order microphones as a function of the dimensionless parameter kd, is shown in Figure 2.27(a). We note here that of all of the microphone designs discussed, the hypercardioid design has the lower K factor than a dipole. This is in apparent contradiction to other superdirectional array designs that can be found in the literature [5]. Typically, higher directional gain results in a higher the value of K. The reason for the apparent contradiction in Fig. 2.27(a) is that the overall gain of the hypercardioid

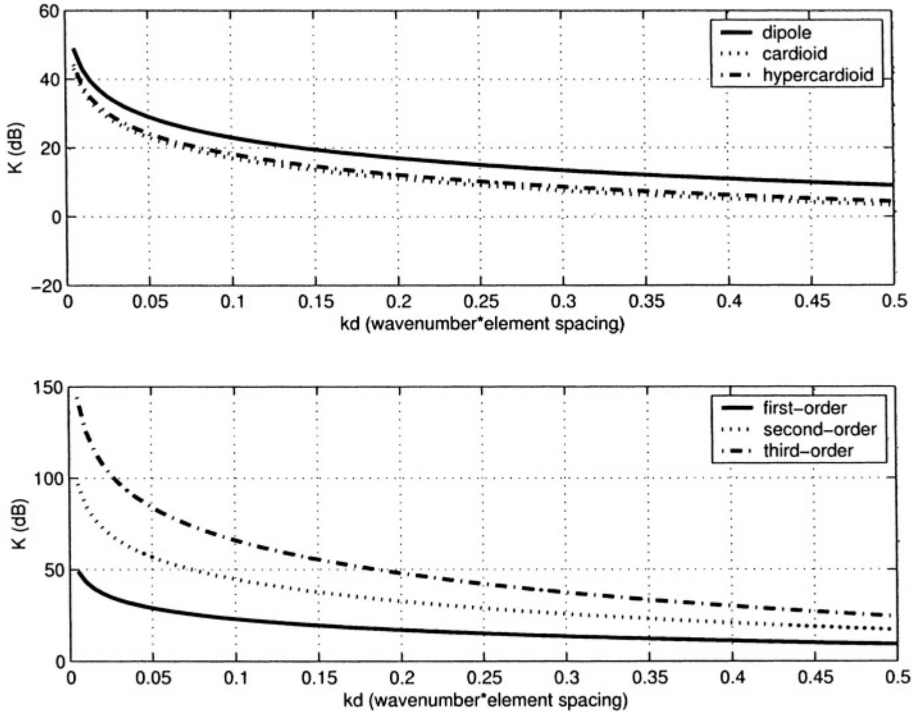


Figure 2.27 Sensitivity as a function of wavelength element-spacing product for, (a) various first-order differential microphones, and, (b) first, second, and third-order dipoles.

is higher than the dipole microphone shown since the delay increases the array output. Other first-order designs with lower values of K are possible, but these do not exhibit the desired optimum directional patterns. Figure 2.27(b) shows the sensitivity function for first, second, and third-order dipoles. It is obvious that a higher order differential array has a larger value of K. Also, as the phase delay (d/c) between the elements is increased, the upper frequency limit of the usable bandwidth is reduced.

Another problem that is directly related to the sensitivity factor K is the susceptibility of the differential system to microphone and electronic preamplifier noise. If the noise is approximately independent between microphones, the SNR *loss* will be proportional to the sensitivity factor K. At low frequencies and small spacings, the signal to noise ratio can easily become less than 0 dB and that is of course not a very useful design.

As an example, consider the case of a first-order dipole array with an effective dipole spacing of 1 cm. Assume that the self-noise of the microphone is equal to an equivalent sound pressure level of 35 dB re 20μ Pa, which is typical for available first-order differential microphones. Now, we place this first-order differential dipole at 1 meter from a source that generates 65 dB at 500 Hz at 1 meter (typical of speech levels). The resulting first-order differential microphone output SNR from Fig. 2.27(b), is only 9 dB. For a second-order array with equivalent spacing the SNR would be –12 dB, and for a third-order array, –33 dB. Although this example makes differential arrays more than first-order look hopeless, there are design solutions that can improve the situation.

In the design of second-order arrays, West, Sessler, and Kubli [15] used baffles around first-order dipoles to increase the second-order differential signal-to-noise. The diffraction caused by the baffle effectively increases the dipole distance d, by a factor that is proportional to the baffle radius. The diffraction is angle and frequency dependent and, if used properly, can be exploited to offer superior performance to an equivalent dipole composed of two zero-order (omnidirectional) microphones. The use of the baffles discussed in reference [15] resulted in a an effective increase in the SNR by approximately 10 dB. The benefit of the baffles used by West, Sessler, and Kubli, becomes clear by examining (2.127) and noting the aforementioned increase in the effective dipole distance.

Another possible technique to improve both the signal-to-noise ratio and reduce the sensitivity to microphone amplitude and phase mismatch, is to split the design into multiple arrays: each covering a specific frequency range. In the design of a differential array, the spacing must be kept small compared to the acoustic wavelength. Since the acoustic wavelength is inversely proportional to the frequency, the desired upper frequency cutoff for an array sets the array microphone spacing requirements. If we divide the differential array into frequency subbands, then the ratio of upper frequency to lower frequency cutoff can be reduced and the spacing for each subband can be made much larger than the spacing for a full-band differential array. The increase in signal-to-noise ratio is proportional to the relative increase in spacing allowed by the use of the subband approach. If the desired frequency range is equally divided into M subbands, the lowest subband SNR increase will be proportional to $20\log_{10}(M)$. The increase in SNR for each increasing frequency subband will diminish until the highest subband which will have the same SNR as the full-band system. The subband solution does have some cost: the number of array elements must also increase. The increase is at least m, and by reuse of the array elements in the subband arrays, can be controlled to be less that nm, where n is the differential array order.

Finally, another approach to control microphone self-noise would be to construct many differential arrays that are very close in position to each other. By combining the outputs of many arrays with uncorrelated self-noise, the SNR can be effectively enhanced by $10\log_{10}(M)$, where M is the number of individual differential arrays.

7. CONCLUSIONS

Information regarding the design and analytical development of optimal differential microphones can not be found in the literature. The purpose this chapter was to provide a basis for the design of differential microphone array systems. Systems with differential orders greater than three require microphones and calibration with tolerances and noise levels not yet available with current electroacoustic transducers. Higher order systems are also somewhat impractical in that the relative gain in directivity is small $[O(\log_{10}(n))]$ as the order n of the microphone increases. Differential microphone array designs are primarily limited by the sensitivity to microphone mismatch and self-noise. The results for many of the differential microphone array designs discussed in this chapter are summarized in Table 2.6.

References

[1] M. Abramowitz and I. A. Stegun, *Handbook ofMathematical Functions.* Dover, New York, 1965.

[2] B. R. Beavers and R. Brown, “Third-order gradient microphone for speech reception,” *J. Audio Eng. Soc.,* vol. 16, pp. 636-640, 1970.

[3] D. K. Cheng, “Optimization techniques for antenna arrays,” *Proc. IEEE,* vol. 59, pp. 1664-1674, Dec. 1971.

[4] L. J. Chu, “Physical limitations of omni-directional antennas,” *J. Applied Physics,* vol. 19, pp. 1163-1175, 1948.

[5] H. Cox, R. Zeskind, and T. Kooij, “Practical supergain,” *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-34, pp. 393-398, June 1986.

[6] C. L. Dolph, “A current distribution for broadside arrays which optimizes the relationship between beamwidth and sidelobe level,” *Proc. IRE,* pp. 335-348, June 1946.

[7] E. N. Gilbert and S. P. Morgan, “Optimum design of directive antenna array subject to random variations,” *Bell Syst. Tech. J.,* vol. 34, pp. 637-663, 1955.

[8] R. F. Harrington, “On the gain and beamwidth of directional antennas,” *IRE Trans. Anten. and Prop.,* pp. 219-223, 1958.

[9] V. I. Korenbaum, “Comments on unidirectional, second-order gradient microphones,” *J. Acoust. Soc. Am.,* 1992.

[10] Y. T. Lo, S. W. Lee, and Q. H. Lee, “Optimization of directivity and signal-to-noise ratio of an arbitrary antenna array,” *Proc. IEEE,* vol. 54, pp. 1033-1045, 1966.

[11] R. N. Marshall and W. R. Harry, “A new microphone providing uniform directivity over an extended frequency range,” *J. Acoust. Soc. Am.,* vol. 12, pp. 481-497, 1941.

[12] H. F. Olson, *Acoustical Engineering.* D. Van Nostrand Company, Inc., Princeton, NJ, 1957.

[13] A. T. Parsons, “Maximum directivity proof of three-dimensional arrays,” *J. Acoust. Soc. Amer.,* vol. 82, pp. 179-182, 1987.

[14] M. R. Schroeder, “The effects of frequency and space averaging on the transmission responses of mulitmode media,” *J. Acoust. Soc. Am.,* vol. 49, pp. 277-283, 1971.

[15] G. M. Sessler and J. E. West, "Second-order gradient unidirectional microphone utilizing an electret microphone," *J. Acoust. Soc. Am.,* vol. 58, pp. 273-278, July 1975.

[16] C. T. Tai, "The optimum directivity of uniformly spaced broadside arrays of dipoles," *IEEE Trans. Anten. and Prop.*, pp. 447-454, 1964.

[17] A. I. Uzkov, "An approach to the problem of optimum directive antenna design," *Compt. Rend. Dokl. Acad. Sci. USSR,* vol. 53, pp. 35-38, 1946.

[18] D. E. Weston, "Jacobi arrays and circle relationships," *J. Acoust. Soc. Amer.,* vol. 34, pp. 1182-1167, 1986.

This page intentionally left blank

Chapter 3

SPHERICAL MICROPHONE ARRAYS FOR 3D SOUND RECORDING

Jens Meyer
mh acoustics
jmm@mhacoustics.com

Gary W. Elko
Avaya Labs, Avaya
gwe@avaya.com

Abstract With the recent widespread availability of inexpensive DVD players and home theater systems, surround sound has become a mainstream consumer technology. The basic recording techniques for live sound events have not changed to accommodate this new dimension of sound field playback. More advanced analysis of sound fields and forensic capture of spatial sound also require new microphone array systems. This chapter describes a new spherical microphone array that performs an orthonormal decomposition of the sound pressure field. Sufficient order decomposition into these eigenbeams can produce much higher spatial resolution than traditional recording systems, thereby enabling more accurate sound field capture. A general mathematical framework based on these eigenbeams forms the basis of a scalable representation that enables one to easily compute and analyze the spatial distribution of live or recorded sound fields. A 24 element spherical microphone array composed of pressure microphones mounted on the surface of a rigid spherical baffle was constructed. Experimental results from a real-time implementation show that a theory based on spherical harmonic eigenbeams matches measured results.

Keywords: Microphone Array, Beamforming, Spherical, 3D Sound Recording

1. INTRODUCTION

A microphone array typically consists of two units: an arrangement of two or more microphones and a beamformer that linearly combines the microphone signals. This combination allows picking up sound signals dependent on their direction of propagation. Their advantage over conventional directional microphones, like a shotgun microphone, is their high flexibility due to the degrees of freedom offered by the multitude of elements and the associated beamformer. The directional pattern of a microphone array can be varied over a wide range. This can be done by changing the beamformer, which typically is implemented in software. Therefore no mechanical alteration of the system is needed.

There are several standard array geometries of which the most common is a linear array. An advantage of using a uniformly spaced linear array is its simplicity with respect to analysis which is equivalent to FIR filter design. This chapter describes a spherical array geometry which has several advantages over other geometries: the beampattern can be steered to any direction in 3-D space without changing the shape of the pattern and the spherical array allows full 3-D control of the beampattern.

An early publication by DuHamel [1] in 1952 described a design approach using a spherical harmonic expansion for narrow-band spherical beamforming antenna arrays. Later work on spherical antenna arrays can be found in e.g. [2, 3, 4]. These papers apply *standard* beamforming techniques, known for example from linear arrays. Perhaps the first quasi-spherical microphone array was patented by Craven and Gerzon [5] which consisted of four directional microphones placed on a virtual sphere at the corners of a tetrahedron. This array allows the recording of zero and first-order spherical harmonics and has been used for Ambisonics recordings [6]. More recently there has been a growing interest in spherical microphone arrays [7, 8, 9, 10, 11].

This chapter presents a general beamformer design approach based on spherical harmonics. The array consists of pressure sensors that are located on the surface of a rigid sphere. It is shown that this arrangement brings several advantages: a) the scattering effects of the sphere are rigorously calculable, b) the diffraction of the sphere brings a signal-to-noise (SNR) improvement at low frequencies and c) due to the diffraction and scattering introduced by the rigid sphere, the array is able to pick up all spherical harmonic modes over a wide frequency range.

The goal is to design an array that is capable of recording spherical harmonics of third-order or higher. Incorporating higher-order spherical harmonic modes, significantly increases the spatial resolution and the degrees of freedom for beampattern design. The proposed beamformer consists of two main units: the *eigenbeamformer*, which performs a spatial decomposition of the sound-field into spherical harmonics, and the *modal-beamformer*, which forms the output beam by appropriately combining the spherical harmonics. This segmentation allows a decoupling of the actual beamformer from the sensor locations. It is shown that this structure results in an efficient and elegant beamformer architecture.

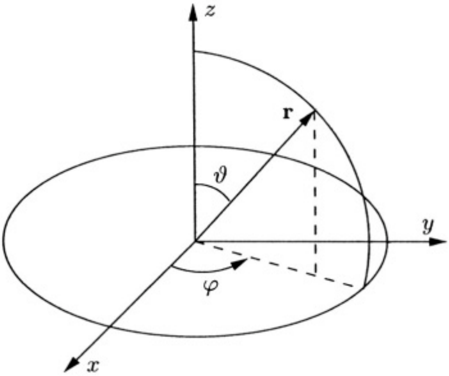


Figure 3.1 Definition of the spherical coordinate system.

Spherical array beamforming can be applied to a wide range of applications such as directional sound pick-up, sound field analysis and reconstruction, passive acoustic tracking, *forensic beamforming*, or room acoustic measurements.

2. FUNDAMENTAL CONCEPT

The main design goal is to record the temporal and spatial information of a sound-field at the position of the array. According to the Helmholtz equation [12] a sound field is uniquely determined if the sound pressure and particle velocity are known on a closed surface. Knowledge of these quantities allows one to calculate the sound-field inside and outside of this surface as long as no sources or obstacles are in the reconstructed volume. Therefore it follows that one needs to measure the sound pressure and the particle velocity only on a closed surface in order to record all information of the surrounding sound-field. To greatly simplify the array construction, only acoustic pressure-sensing microphones are integrated into the surface of a rigid sphere are used. Using a rigid body has the advantage that the radial particle velocity on its surface is zero. This greatly simplifies the general solution since only the sound pressure needs to be measured. The spherical shape was chosen to keep the mathematics as simple as possible and, from symmetry, put equal weight on all directions. Theoretically many other shapes are possible.

To keep this chapter self-contained, a brief review of the physical principle of the spherical array is presented, that is based on a rigid spherical scatterer. For more detailed analysis, the reader is referred to e.g. [13]. Figure 3.1 shows the notation for the spherical coordinate system used throughout this chapter.

A plane-wave impinging at an angle ϑ relative to the z-direction can be expressed in spherical coordinates as follows [13]:

$$\begin{aligned} G(kr, \vartheta, t) &= e^{i(\omega t + kr\cos\vartheta)} \\ &= \sum_{n=0}^{\infty}(2n+1)i^n j_n(kr)P_n(\cos\vartheta)e^{i\omega t}. \end{aligned} \tag{3.1}$$

To keep the exposition compact, the results given in this chapter are restricted to plane-wave incidence. If desired, results can be extended to spherical wave incidence [13]. In (3.1), j_n represents the spherical Bessel function of the first kind of order n. The letter i is used for the imaginary constant to avoid confusion with the spherical Bessel function. P_n is the Legendre function of order n and degree m, and k represents the wave-number which is equal to $2\pi/\lambda$ where λ is wavelength.

From (3.1), the sound particle velocity for an impinging plane-wave on a spherical surface can be derived using Euler's equation. If the spherical surface is rigid, the sum of the radial velocity of the incoming and the scattered sound wave has to be zero on this surface. Using this boundary condition, the reflected sound pressure can be determined and the resulting sound pressure field becomes the superposition of the incoming and the scattered sound pressure field:

$$G(kr, ka, \vartheta) = \\ \sum_{n=0}^{\infty}(2n+1)i^n\left(j_n(kr) - \frac{j_n'(ka)}{h_n^{(2)'}(ka)}h_n^{(2)}(kr)\right)P_n(\cos\vartheta). \tag{3.2}$$

The prime denotes the derivative with respect to the argument and a is the radius of the sphere while $h_n^{(2)}$ is the spherical Hankel function of the second kind of order n. From this point, the time dependence is omitted for better readability of the equations.

To find a more general expression that gives the sound pressure at a point $[r_s, \vartheta_s, \varphi_s]$ for an impinging sound wave from direction $[\vartheta, \varphi]$, the following Legendre addition theorem is essential [14]:

$$P_n(\cos\Theta) = \sum_{m=-n}^{n}\frac{(n-m)!}{(n+m)!}P_n^m(\cos\vartheta)P_n^m(\cos\vartheta_s)e^{im(\varphi-\varphi_s)}, \tag{3.3}$$

where Θ is the angle between the direction of the impinging sound wave $[\vartheta, \varphi]$ and the radius vector of the observation point $[r_s, \vartheta_s, \varphi_s]$. Substituting (3.3) into (3.2) gives the following solution:

$$G(kr_s, \vartheta_s, \varphi_s, ka, \vartheta, \varphi) =$$

$$4\pi \sum_{n=0}^{\infty} i^n b_n(ka, kr_s) \sum_{m=-n}^{n} Y_n^m(\vartheta, \varphi) Y_n^{m^*}(\vartheta_s, \varphi_s), \tag{3.4}$$

where the asterisk $(\cdot)^*$ denotes the complex conjugate operation. In (3.4), two abbreviations, b_n and Y_n^m, are introduced:

$$b_n(ka, kr_s) = j_n(kr_s) - \frac{j_n'(ka)}{h_n^{(2)'}(ka)} h_n^{(2)}(kr_s), \tag{3.5}$$

$$Y_n^m(\vartheta, \varphi) = \sqrt{\frac{(2n+1)}{4\pi} \frac{(n-m)!}{(n+m)!}} P_n^m(\cos\vartheta) e^{im\varphi}. \tag{3.6}$$

These quantities play a major role in the concept of the spherical array. In the following, b_n will be referred to as *modal strength* or *modal coefficient*, Y_n^m represents the spherical harmonic of order n and degree m. One important property is their mutual orthonormality:

$$\int_0^{2\pi} \int_0^{\pi} Y_n^m(\vartheta, \varphi) Y_{n'}^{m'^*}(\vartheta, \varphi) \sin\vartheta \, d\vartheta \, d\varphi = \delta_{nn'} \delta_{mm'}. \tag{3.7}$$

This property makes spherical harmonics very attractive to beamforming. Basically any square integrable function defined on the surface of a sphere, e.g. a beampattern, can be expanded into a spherical harmonic series. In fact, some beamforming work based on spherical harmonics has been done in the past [15]. Equation (3.4) represents a key result of this analysis. It allows the representation of a plane-wave sound-field as an expansion of spatially orthonormal spherical harmonic components. Since a far-field sound field can be modeled as a superposition of multiple plane-waves, (3.4) allows an expansion of any far-field sound-field into an orthonormal series of spherical harmonics. In the following we will refer to these orthonormal components as modes of order n and degree m.

Based on (3.4), the next section describes the eigenbeamformer which decomposes the sound-field into the orthonormal modes while Section 4 describes the modal beamformer which combines these modes to realize desired beampatterns.

3. THE EIGENBEAMFORMER

As shown in Figure 3.2, the eigenbeamformer can be logically divided into to separate cascaded sections. The first step of this two-stage beamforming process can be viewed as a preprocessor to the actual forming of the output

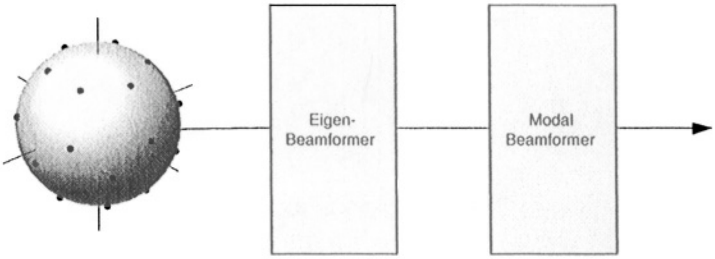


Figure 3.2 Block diagram of the functional blocks of the spherical array.

beam which is done in the modal-beamformer second stage. As will be shown in Section 4, introducing this preprocessor step results in many advantages for the beamforming itself.

The task of this preprocessor is to transform the microphone signals into an orthogonal beam-space. Since these beams are characteristics of the sound-field in a similar way as eigenvectors are for a matrix, they are referred to as eigenbeams. Hence, the preprocessor is referred to as an eigenbeamformer.

The decomposition of the sound-field is based on the orthonormal property of the spherical harmonics. To begin, assume a rigid sphere covered with a continuous pressure sensitive surface and the sensitivity of this surface be position-dependent and described by the spherical harmonic $Y_{n'}^{m'}(\vartheta_s, \varphi_s)$. The output of such a microphone is:

$$\begin{aligned} F_{n',m'}(\vartheta, \varphi, ka, kr_s) &= \int_{\Omega_s} G(kr_s, \vartheta_s, \varphi_s, ka, \vartheta, \varphi) Y_{n'}^{m'}(\vartheta_s, \varphi_s)\, d\Omega_s \\ &= 4\pi i^{n'} b_{n'}(ka, kr_s) Y_{n'}^{m'}(\vartheta, \varphi), \qquad (3.8) \end{aligned}$$

where $d\Omega_s$ represents an integration over the surface of the sphere. This result states that the far-field directivity of the microphone has the same directional dependance as its surface sensitivity function, namely $Y_{n'}^{m'}$. The factor $4\pi i^n$ introduces a phase shift and scaling that can be easily compensated and is neglected in the further discussions. Another factor is the modal coefficient b_n. It introduces a frequency dependance that must taken into account and is further investigated in Section 3.3. Yet another problem to be solved is the change from the continuous microphone aperture as it is used in (3.8) to a sampled aperture for the spherical array. This step is necessary since a) a position dependent continuous sensitivity would be extremely difficult to manufacture and b) a separate sphere for every eigenbeam would be required. The step from a continuous to a discrete aperture is described in the next section.

3.1 DISCRETE ORTHONORMALITY

To make a practical array, the continuous aperture needs to be sampled. To achieve the same result as obtained in (3.8) the sample positions have to fulfill the following *discrete orthonormality* condition:

$$A_{nm} \sum_{s=0}^{S-1} Y_n^m(\vartheta_s, \varphi_s) Y_{n'}^{m'}(\vartheta_s, \varphi_s) = \delta_{nn'}\delta_{mm'}. \tag{3.9}$$

Note that the subscript s was previously used to identify a point on the spherical surface while in (3.9) it enumerates the sensors that are located on the surface. It is a difficult task to find a set of sensor locations that fulfill the orthonormality. To relax the constraint from orthonormality to orthogonality, a factor A_{nm} is introduced. This factor can be further reduced by defining:

$$A_n = a_n \frac{4\pi}{S}. \tag{3.10}$$

The factor 4π becomes necessary since there is no integration over a sphere for the discrete case where it becomes a sum over S sensors which are normalized by the factor $1/S$. Instead of including this re-normalization in an additional factor A_n, this new normalization can be included in the spherical harmonics immediately. To avoid unnecessary confusion this approach is not pursued here.

One sensor arrangement that fulfills the constraint of discrete orthonormality up to modes of 4th-order is the center of the 32 faces of a truncated icosahedron. Another arrangement using only 24 sensors was found, which achieves orthogonality up to 3rd-order modes (see Section 9).

The resulting structure of the eigenbeamformer can be derived from (3.9): for a specific eigenbeam Y_n^m the microphone signals are first weighted by the sampled values of the corresponding surface sensitivity, $Y_n^m(\vartheta_s, \varphi_s)$, then Corrected by A_{nm}, and finally summed. The outputs of this beamformer are the eigenbeams.

Besides the orthonormal constraint, spatial aliasing has to be considered when sampling a discrete aperture. Just as sampling a time waveform requires a minimum number of samples per time interval in order to be able to recover the original signal, sampling the spatial aperture requires a minimum number of sample locations to recover the original spatial signal distribution. Since there are $(N+1)^2$ spherical harmonics for a spatial resolution of order N (see Section 3.2), a minimum of $(N+1)^2$ sample locations are required to distinguish the spherical harmonics.

3.2 THE EIGENBEAMS

The outputs of the eigenbeamformer are a set of orthonormal beam-patterns, the eigenbeams. These eigenbeams represent a spatially orthonormal decom-

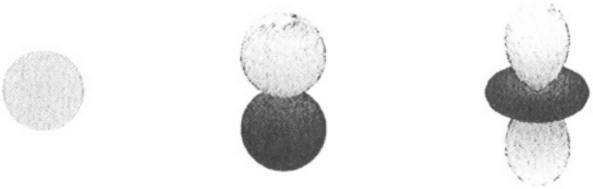


Figure 3.3 Eigenbeams of order 0, 1, and 2 (degree 0).

position of the sound-field, as shown in (3.4). A complete set of eigenbeams contains all spatial information about the original sound-field.

Figure 3.3 shows some example eigenbeams. From (3.6) it can be seen that the elevation dependance follows the Legendre function while the azimuth dependence has a sine-cosine dependance. The order n determines the number of zeros in ϑ-direction while two times the degree m gives the number of zeros in the φ-direction.

The number of eigenbeams depends on the desired spatial resolution for the application. The total number of eigenbeams up to N-th order is $(N+1)^2$. For example, in a directional microphone application the maximum achievable directional gain is $20\log_{10}(N+1)$. To obtain a maximum directional gain of 12 dB for an arbitrary direction, one would need all eigenbeams up to third-order, which is 16 eigenbeams. As mentioned before, the number of microphones needs to be equal or larger than the number of eigenbeams. The example assumes full 3D coverage. If one is interested only in the horizontal spatial resolution the number of eigenbeams can be reduced significantly to $2N+1$.

The beampattern of the eigenbeams is frequency independent. However the magnitude shows a dependance according to the modal coefficient. This dependence will be analyzed in the next section

3.3 THE MODAL COEFFICIENTS

From (3.8), it is seen that the eigenbeams exhibit a frequency dependence according to the modal coefficient b_n. The magnitude of the frequency response for these coefficient is plotted in Fig. 3.4 for various orders n.

In Fig. 3.4, it can be seen that at very low frequencies the zero-th order mode is dominant. For $ka = 0.2$ (for a sphere of radius 5cm, this would result in a frequency of 220 Hz), the first-order mode is down by 20 dB. At higher frequencies more modes emerge. The rising slope of the modal coefficients is $6N$ dB per octave. Once a mode has reached an adequate level, it can be used for further processing. The level depends on the desired SNR of the overall system.

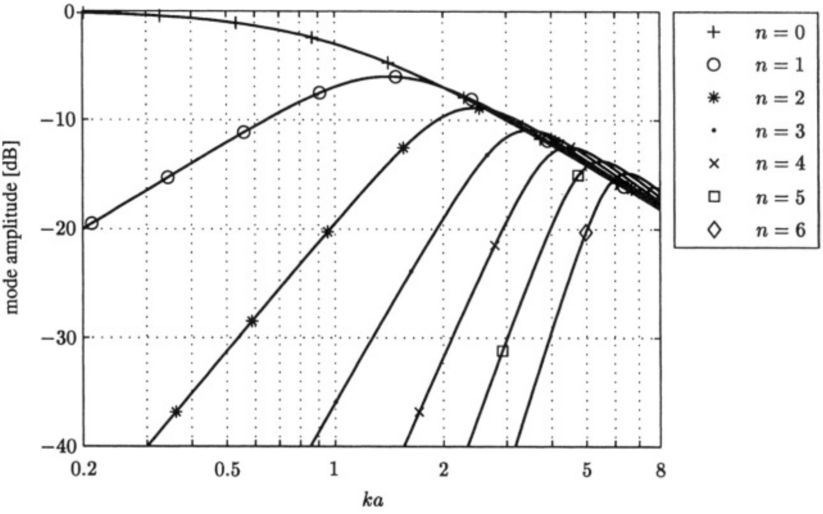


Figure 3.4 **Mode coefficients b_n for different orders.**

To allow subsequent combination of the eigenbeams they should have a flat frequency response. This means that the output of the eigenbeamformer should be filtered with the inverse frequency response of the modal coefficients. For low frequencies this is basically an amplification. Depending on the performance of the microphones and associated hardware and the desired SNR, the maximum gain needs to be limited to a certain threshold. The result is that each eigenbeam has a low frequency cutoff below which the eigenbeam should not be used for further processing.

The sound field around the sphere contains modes of higher orders than the array can sample. For example, at 5 kHz the sound-field around a sphere of 5 cm radius $(ka = 4.6)$ contains spherical harmonics of significant strength up to fifth-order. A spherical array with 32 microphones is able to handle spherical harmonics up to fourth-order. To enable a wideband response while avoiding aliasing, one has to provide a spatial low-pass filter. Such a low-pass filter can be implemented using microphones with large membranes or *patch-microphones.* The term patch-microphone refers to microphones that cover a continuous section of the spherical surface as opposed to point microphones. By integrating the sound over a large area, higher order modes will be attenuated. Such a patch microphone might be built by using pressure sensitive materials that can be placed conformingly onto the surface of a sphere or made by combining many closely-spaced pressure microphones.

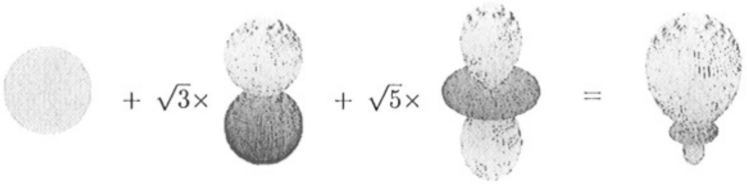


Figure 3.5 Illustration of generating a second order hypercardioid pattern. (Note that only the directional properties are shown. The magnitude is scaled to unity for the look direction.)

4. MODAL-BEAMFORMER

The modal beamformer forms the second stage of the overall array processing structure. The name *modal beamformer* was chosen to emphasize its difference to a *conventional* beamformer: typically the input signals to a beamformer are the microphone signals. However, the modal beamformer takes spatial modes, the eigenbeams as input signals. This design approach allows a very simple and powerful design that consists of the actual beam shaping unit (combining unit) and the steering unit. Both units are independent of each other allowing a change of the beam-pattern while maintaining the look-direction or maintaining the beam-pattern while steering the look-direction.

4.1 COMBINING UNIT

The combining unit is a simple weight-and-sum beamformer:

$$d(\vartheta) = \sum_{n=0}^{N} c_n Y_n(\vartheta, \varphi), \tag{3.11}$$

where the beamformer multiplies each input beam by a factor c_n and sums all weighted beams. As an example, Fig. 3.5 shows the generation of a second-order hypercardioid pattern steered along the z-axis. It can be shown that the weights for a hypercardioid pattern are: $c_0 = 1$, $c_{10} = \sqrt{3}$, and $c_{20} = \sqrt{5}$. Beams with degree not equal to zero are weighted with zero.

Using only zero-degree eigenbeams limits the beampattern in this design stage to be φ-independent. However, is is shown in the next section that this greatly simplifies the steering of the pattern.

4.2 STEERING UNIT

From Section 4.1, the beampattern d is obtained according to (3.11). Using (3.6), this can be rewritten as:

$$d(\vartheta) = \sum_{n=0}^{N} c_n \sqrt{\frac{2n+1}{4\pi}} P_n(\cos\vartheta). \tag{3.12}$$

To steer a beampattern towards a desired look-direction $[\vartheta_0, \varphi_0]$, (3.3) can be substituted into (3.12) which results in the new coefficients $\hat{c}_{nm}$:

$$\hat{c}_{nm} = c_n \sqrt{\frac{(n-m)!}{(n+m)!}} P_n^m(\cos\vartheta_0) e^{-im\varphi_0}. \tag{3.13}$$

Equation (3.13) shows that the steering and beam-shaping coefficients are connected in a multiplicative manner. To simplify the overall structure of the modal-beamformer, one fact that can be exploited is that the steering related terms are applied to all eigenbeams of a given order, while the weighting coefficient c_n only needs to be applied on a per order basis. To separate the steering and the beam-shaping, we can rewrite (3.11) using (3.13):

$$d(\vartheta, \varphi) = \underbrace{\sum_{n=0}^{N} c_n}_{combine} \underbrace{\sum_{m=-n}^{n} \sqrt{\frac{(n-m)!}{(n+m)!}} P_n^m(\cos\vartheta_0) e^{-im\varphi_0}}_{steer}. \tag{3.14}$$

5. ROBUSTNESS MEASURE

An important characteristic of a microphone array is its sensitivity to deviations from the ideal implementation. These deviations include: a) errors in the sensor locations, b) variations in amplitude and phase, and c) sensor self noise. A common measure for these non-ideal limitations is the noise sensitivity [16] or its inverse, the white-noise-gain (WNG). In this chapter the WNG will be used. The WNG is a measure of the robustness of the array, meaning that the higher the WNG of an array, the more robust it is against the above mentioned errors occurring in practical implementations. It is defined as:

$$\text{WNG}(\omega) = \frac{|d(\vartheta_0, \varphi_0, \omega)|^2}{\sum_{s=0}^{S-1} |H_s(\omega)|^2}, \tag{3.15}$$

where d represents the array output for the look-direction $[\vartheta_0, \varphi_0]$ and H_s is the array filter for sensor s. The numerator can be interpreted as the signal energy at the output of the array, while the denominator is the sensor self noise power. The sensor noise is assumed to be independent from senor to sensor. It may not be immediately obvious that this measure also quantifies the sensitivity of the array to errors in the setup. For more details see reference [16].

The goal of this section is to find some general approximations for the WNG that allow an estimate of the robustness of the spherical array. To simplify the notation without loss of generality, the look-direction is assumed to be in the z-direction.

To find an expression for the spherical array in the numerator of (3.15), the array output d can be replaced by (3.11). The array filters H_s in the denominator of (3.15) can be replaced by the following expression:

$$H_s(\omega) = 4\pi \sum_{n=0}^{N} \frac{c_n(\omega)}{i^n b_n(\omega)} a_n \frac{4\pi}{S} Y_n(\vartheta_s, \varphi_s). \tag{3.16}$$

Substituting d and H_s into (3.15), the WNG for the spherical array becomes:

$$\mathrm{WNG}(\omega) = S^2 \frac{\left| \sum_{n=0}^{N} c_n Y_n(\vartheta_0, \varphi_0) \right|^2}{\sum_{s=0}^{S-1} \left| \sum_{n=0}^{N} \frac{c_n(\omega) a_n}{i^n b_n(\omega)} Y_n(\vartheta_s, \varphi_s) \right|^2}. \tag{3.17}$$

From (3.17) it can be seen that the WNG depends to a large extent on the sampling scheme of the sphere. This makes a general prediction difficult. However, the following two special cases are investigated: a) the WNG of an individual eigenbeam and b) a super-directional pattern with $b_n \ll b_{n-1}$.

For a single eigenbeam the WNG from (3.17) becomes:

$$\mathrm{WNG}_n(\omega) = \frac{S^2 \left| b_n(\omega) \right|^2 \left| Y_n(\vartheta_0, \varphi_0) \right|^2}{\left| a_n \right|^2 \sum_{s=0}^{S-1} \left| Y_n(\vartheta_s, \varphi_s) \right|^2}. \tag{3.18}$$

Again this expression depends on the sensor locations. However, for the eigenbeam of order zero a simplified expression can be found:

$$\mathrm{WNG}_0(\omega) = S \left| b_0(\omega) \right|^2. \tag{3.19}$$

From Fig. 3.4, it can be seen that the modal coefficient for the zero-order mode equals unity for low frequencies. Therefore we find a WNG of S for the zero-order mode, which is the well known result for the maximum WNG achievable with an array. Towards higher frequencies b_0 decreases and so does the WNG. This is different from a delay-and-sum beamformer where the maximum WNG remains S. The reason for the decrease is that in this analogy we are only looking at the zero-order mode. With increasing frequency the sound energy is distributed over an increasing number of modes. If only the zero-order mode is used, this additional energy in the higher-order modes requires a proportional loss in energy in the zero-th order mode resulting in a decrease in the WNG.

The second special case is a superdirectional pattern for which $b_n \ll b_{n-1}$. Using this constraint (3.17) becomes:

$$\mathrm{WNG}_N(\omega) = S^2 \left| \frac{b_N(\omega)}{c_N(\omega) a_N} \right|^2 \frac{\left| \sum_{n=0}^{N} c_n Y_n(\vartheta_0, \varphi_0) \right|^2}{\sum_{s=0}^{S-1} \left| Y_N(\vartheta_s, \varphi_s) \right|^2}. \tag{3.20}$$

From (3.20) it can be seen that as long as the modal beamformer only uses frequency independent weights c_n, the frequency dependance is determined by the modal coefficients b_N. According to Fig. 3.4, this is a 6*N* dB per octave slope. Again, this result agrees with the well known behavior of a differential array [17].

6. BEAMPATTERN DESIGN

Using the eigenbeams as input signals to the beamformer simplifies the beampattern design greatly. This section describes two design concepts: first the design of an arbitrary beam-pattern and second the design of optimum beampattern with regards to the directivity index under the constraint of a minimum WNG.

6.1 ARBITRARY BEAMPATTERN DESIGN

The main design goal is to achieve a desired beampattern $d(\vartheta, \varphi)$. Thus, one needs to find the modal weights c_n. Exploiting the orthonomality property of the spherical harmonics, these weights can easily be found according to:

$$c_{nm} = \int_\Omega d(\vartheta, \varphi) Y_n^m(\vartheta, \varphi) \, d\Omega. \tag{3.21}$$

Theoretically any beam-pattern can be realized. In practice, however, d is limited by the spherical harmonics that are available to the modal beamformer. As discussed earlier, the highest order of a practically typical spherical array will be around fourth-order.

6.2 OPTIMUM BEAMPATTERN DESIGN

This subsection describes a method to compute the coefficients c_n that result in a maximum achievable directivity index (DI). A constraint on the WNG is included in the optimization. The optimization method adapts the approach given by Cox et. al. in [18]. The directivity factor (*D*) of a microphone is defined as the ratio of energy picked up by an omnidirectional microphone to the energy picked up by a directive microphone in an isotropic noise field. Both microphones must have the same sensitivity towards the *look* direction. The DI is 10 times the 10-base logarithm of the directivity factor *D*. If a directive microphone is used in a spherically isotropic noise field, the DI can be seen as the acoustical signal-to-noise (SNR) improvement achieved by the directive microphone for signals propagating along the look direction. For an array *D* can be written in matrix notation:

$$D(\omega_0) = \frac{\mathbf{w}^H \mathbf{G}_0 \mathbf{G}_0^H \mathbf{w}}{\mathbf{w}^H \mathbf{R} \mathbf{w}}, \tag{3.22}$$

where $(\cdot)^H$ denotes Hermitian transpose. On the right side of the equation the frequency dependence is omitted for readability. The vector $\mathbf{w}$ contains the sensor weights at frequency ω_0:

$$\mathbf{w} = \begin{bmatrix} w_0 & w_1 & \cdots & w_{S-1} \end{bmatrix}^T, \tag{3.23}$$

where $(\cdot)^T$ denotes transpose of a vector or a matrix. The sensor weights $\mathbf{w}$ can be expressed in terms of the modal weights c_n as follows:

$$\mathbf{w} = \mathbf{Hc}. \tag{3.24}$$

This is basically (3.11) in matrix notation. The elements of $\mathbf{H}$ are:

$$h_{sn} = \frac{Y_n(\vartheta_s, \varphi_s)}{i^n b_n(kr_s, ka)}. \tag{3.25}$$

$\mathbf{H}$ is an S-by-N matrix. The vector $\mathbf{c}$ contains the spherical harmonic coefficients c_n used for the beampattern design. $\mathbf{G}_0$ in (3.22) represents a vector describing the source array transfer function for the look direction at ω_0. For a pressure sensor close to a rigid sphere these values can be computed from (3.4). The spatial cross-correlation matrix is $\mathbf{R}$. The matrix elements are defined by:

$$r_{pq} = \frac{1}{4\pi} \int_{\Omega} G(\vartheta_p, \varphi_p, \vartheta, \varphi, kr_p, ka) G(\vartheta_q, \varphi_q, \vartheta, \varphi, kr_q, ka)^* d\Omega. \tag{3.26}$$

In Section 5, the WNG was defined and can be rewritten in matrix notation as follows:

$$\text{WNG} = \frac{\mathbf{w}^H \mathbf{P} \mathbf{w}}{\mathbf{w}^H \mathbf{w}}. \tag{3.27}$$

The above equations assume that only the spherical harmonics of degree 0 are used for the pattern. If desired, the equations can be rewritten to include other spherical harmonics. The goal is now to maximize the D with a constraint on the WNG. This is the same as minimizing the following function, where the Lagrange multiplier ϵ is used to include the constraint:

$$\frac{1}{f} = \frac{1}{D} + \epsilon \frac{1}{\text{WNG}}. \tag{3.28}$$

Following the approach in [18], one obtains the following equation that has to be maximized with respect to the coefficient vector $\mathbf{c}$:

$$f(\mathbf{c}) = \frac{\mathbf{c}^H \mathbf{H}^H \mathbf{P} \mathbf{H} \mathbf{c}}{\mathbf{c}^H \mathbf{H}^H (\mathbf{R} + \epsilon \mathbf{I}) \mathbf{H} \mathbf{c}}, \tag{3.29}$$

where $\mathbf{I}$ is the identity matrix. Equation (3.29) is a generalized eigenvalue problem [19]. Since $\mathbf{H}$, $\mathbf{R}$, and $\mathbf{I}$ are of full rank, the solution is the eigenvector corresponding to:

$$\max\left\{ \lambda \left(\left[\mathbf{H}^H (\mathbf{R} + \epsilon \mathbf{I}) \mathbf{H} \right]^{-1} \left(\mathbf{H}^H \mathbf{P} \mathbf{H} \right) \right) \right\}, \tag{3.30}$$

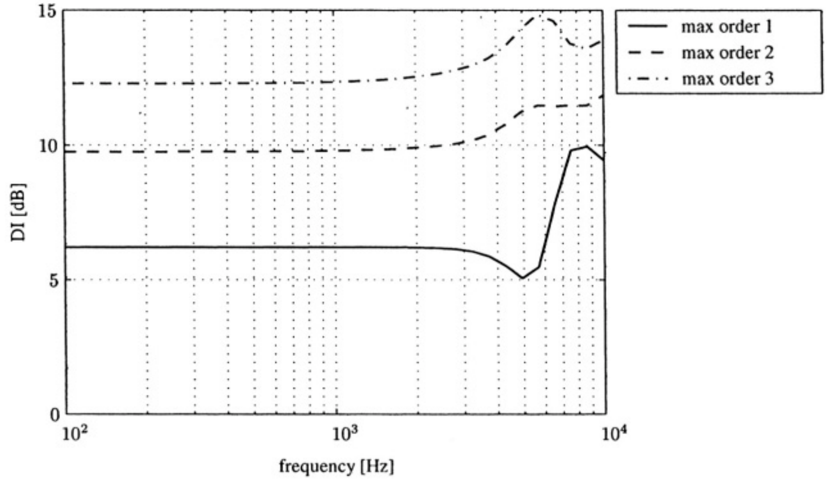


Figure 3.6 **Maximum DI, allowing spherical harmonics up to order N, WNG is arbitrary.**

where $\lambda(\cdot)$ means "eigenvalue from." Unfortunately (3.30) cannot be solved for ϵ. One way to find the maximum D for a desired WNG is as follows:

1. Find the solution to (3.30) for an arbitrary ϵ.
2. From the resulting vector $\mathbf{c}$ compute the WNG.
3. If the WNG is larger than desired, then start again with Step 1, with a smaller ϵ; if the WNG is too small, start again with Step 1, now using a larger ϵ. If the resulting WNG matches the desired WNG, the iteration is complete.

Note that the choice of $\epsilon = 0$ results in the maximum achievable DI. On the other hand, $\epsilon \to \infty$ results in a delay-and-sum beamformer. The latter has the maximum achievable WNG, since all sensor signals will be summed up in phase, yielding the maximum output signal. It can be seen in (3.28) that $f(\mathbf{c})$ depends montonically on ϵ. Figure 3.6 shows the maximum DI that can be achieved with a 24 element array using spherical harmonics up to third-order without a constraint on the WNG. It is well known that the theoretical maximum is $\mathrm{DI}_{\max} = 20\log_{10}(N+1)$ [17]. In Fig. 3.6, it can be seen that there are deviations from the theoretical value at higher frequencies due to spatial aliasing. For a spherical array with 24 elements on the surface of the sphere with $a = 3.75$ cm, the maximum usable frequency is about 5 kHz. This explains the deviations from the theoretical DI starting just below 5 kHz. According to the theoretical limit, the DI for an array using spherical harmonics up to third-order cannot exceed 12 dB. For the given array there will be aliasing

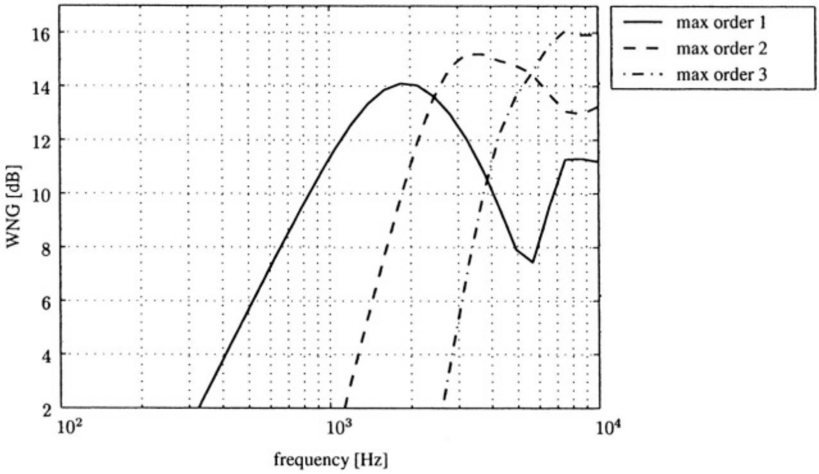


Figure 3.7 WNG corresponding to maximum DI from Fig. 3.6.

starting around 5 kHz, which means that spherical harmonics of higher orders will be included in the beamforming. Therefore a DI larger than 12 dB does not violate the theoretical limit! In Fig. 3.7 one finds the WNG corresponding to the maximum DI in Fig. 3.6. As it was found in Section 5, as long as the pattern is superdirectional, the WNG increases with 6*N* dB per octave. The maximum WNG that can be achieved is about $\mathrm{WNG}_{\mathrm{max}} = 10\log_{10} S$, which is for the 24 element array about 14 dB. In Fig. 3.7, one can see that for the sphere baffled array the maximum WNG is a bit higher, about 16 dB. Once the maximum is reached it decreases. This is due to the fact that the mode number in the array pattern is constant. Since the mode magnitude decreases once a mode has reached its maximum, the WNG is expected to decrease as soon as the highest mode has reached its maximum. For example, the first-order mode shows this for $f = 2$ kHz (compare Fig. 3.4).

Figure 3.8 shows the maximum DI that can be achieved with a constraint on the WNG for a pattern that contains the spherical harmonics up to third-order. Here one can see the tradeoff between WNG and DI. The higher the required WNG, the lower the maximum DI and vice versa. For a minimum WNG of –10 dB one gets a DI of 12 dB above a frequency of about 1.7 kHz. Between 100 Hz and 1.7 kHz the DI increases from 6 dB to 12 dB.

Figures 3.9 and 3.10 give the magnitude and phase of the coefficients computed according to the procedure described above in this section. *N* was set to 3 and the minimum required WNG was –10 dB. The coefficients are normalized so that the sensitivity for the look direction is unity.

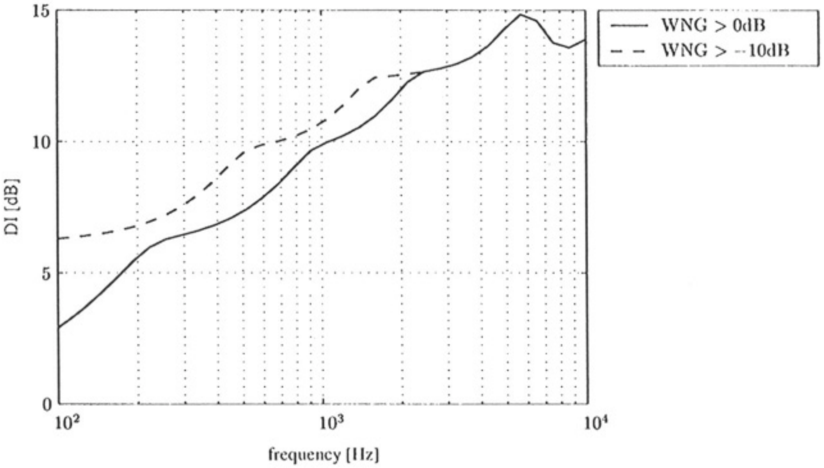


Figure 3.8 Maximum DI with different constraints on the WNG, $N = 3$.

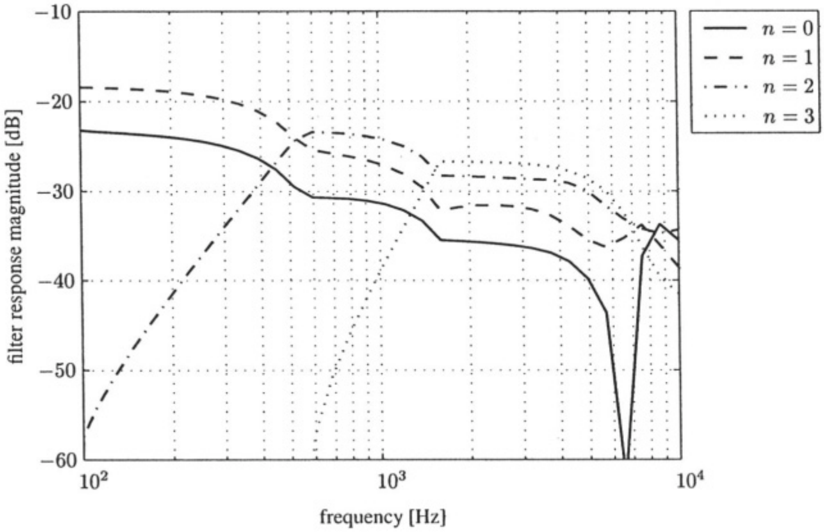


Figure 3.9 Magnitude of filter response $c_n(\omega)$ for maximum DI design with $N = 3$ and WNG ≥ -10 dB.

7. MEASUREMENTS

A 24 element spherical array whose geometry is given in the Appendix, was measured in an anechoic environment. Figures 3.11 to 3.14 show the measured

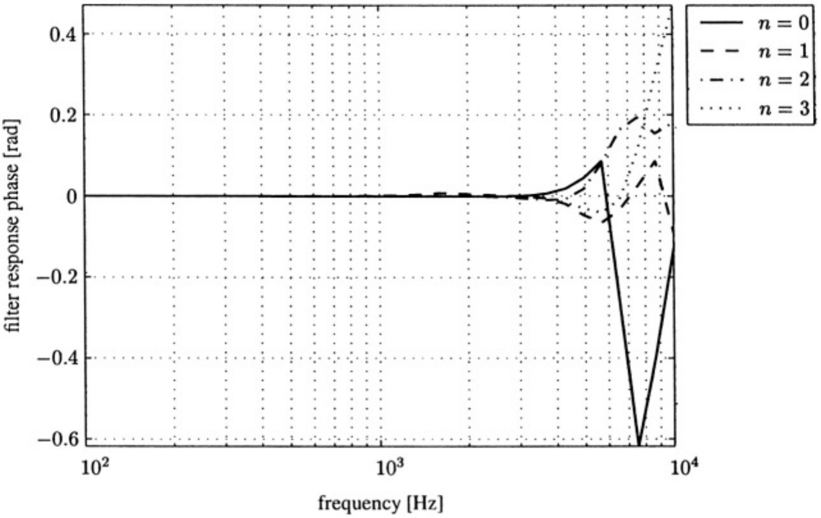


Figure 3.10 Phase of filter response $c_n(\omega)$ for maximum DI design with $N = 3$ and WNG ≥ -10 dB.

beampattern for the real part of the eigenbeams Y_0, Y_1^1, Y_2^2 and Y_3^3. These eigenbeams were chosen since they are the most important for the resolution in the horizontal plane which typically is the preferred plane of operation. The beampatterns are shown in two ways: on the left side the familiar polar plot is shown for selected frequencies; on the right side the pattern is shown over the complete frequency band of operation from 50 Hz to 5 kHz. This allows a better visual display of the frequency dependent behavior of the directional properties. Since we are only interested in directional properties here, the beampatterns are normalized to 0 dB for the look-direction, which is 0° for the eigenbeams of Figs. 3.11 to 3.14. The angular resolution for the measurements was 5°.

It can be seen that the zero and first order beams are almost ideal with only slight deviation from the theoretical pattern. The variations over frequency are very small. For the second- and third-order patterns, significant deviations exist at frequencies below 700 Hz and 1.7 kHz, respectively. This was expected since the WNG is very low in these regions. For a beampattern design this implies that the second-order eigenbeams should not be used below 700 Hz and the third-order eigenbeams not below 1.7 kHz.

Another measurement was made to show an example output of the modal beamformer. An application was used that allows one to vary the directivity (and therefore DI) of the output beam continuously between zero and about 12 dB, the maximum for a third-order system. The beampattern for the medium DI of about 6 dB and for the maximum DI of about 12 dB was measured. It is

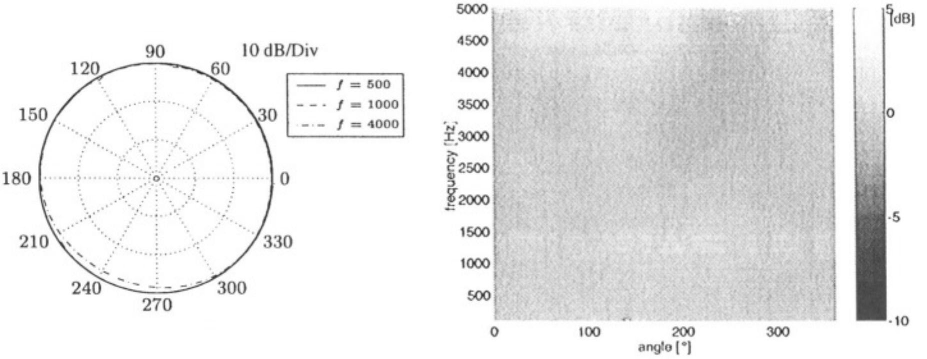


Figure 3.11 Beam pattern of eigenbeam with order zero.

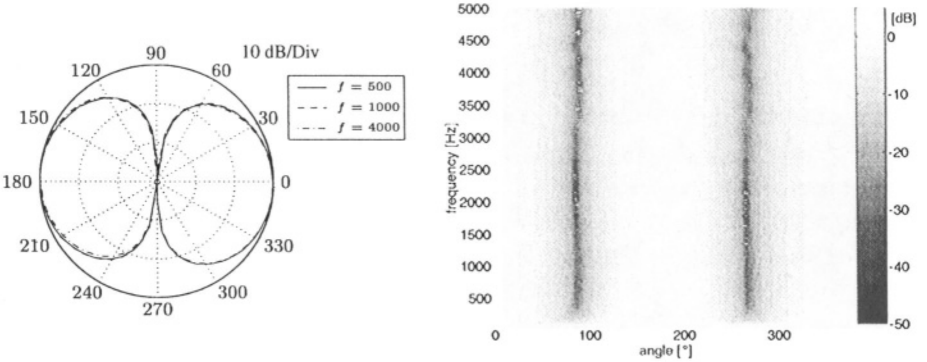


Figure 3.12 Beam pattern of eigenbeam with order one and degree one.

displayed in Figs. 3.15 and 3.16. To show the ability to steer the array, the look-direction is set to 90°. The first pattern is between a first- and second-order cardioid. Since the first-order pattern is dominant, the pattern is frequency invariant over the complete operating range. The second pattern is close to a third-order hypercardioid pattern, which by definition has the highest directivity for any given order. It is interesting to see the transition from a first-order pattern at low frequencies to a second-order pattern at medium frequencies and a third-order pattern at high frequencies. This transition was designed to keep the WNG almost constant over frequencies.

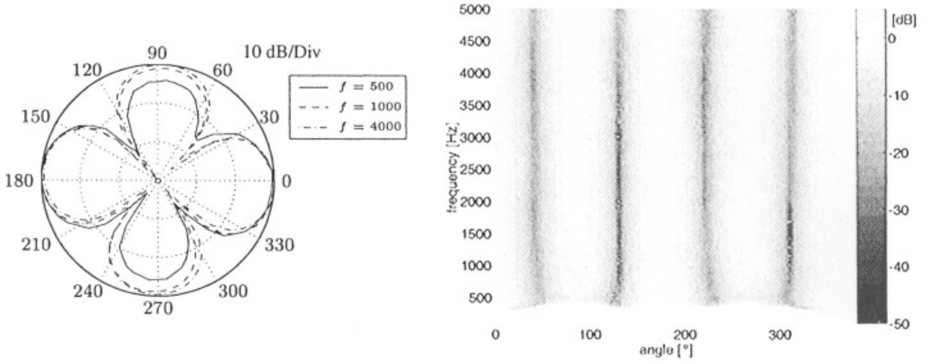


Figure 3.13 Beam pattern of eigenbeam with order two and degree two.

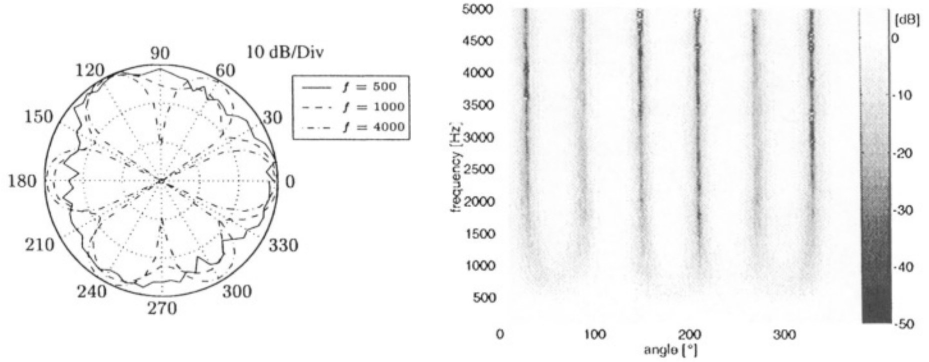


Figure 3.14 Beam pattern of eigenbeam with order three and degree three.

8. SUMMARY

This chapter describes the mathematical framework for a spherical microphone array that is flush mounted on the surface of a rigid sphere. It was shown that the beamforming process can be logically divided into a two-stage beamformer. In the first *eigenbeamformer* stage, the sound-field is decomposed into spatially orthonormal beams which are called eigenbeams. In the second modal-beamformer stage, the beam-shaping and steering is done by a simple and efficient matrix multiplication operation. This two-stage structure results in several advantages, one of which is that the beam-shaping and steering becomes decoupled from the microphone array geometry. The inputs to the

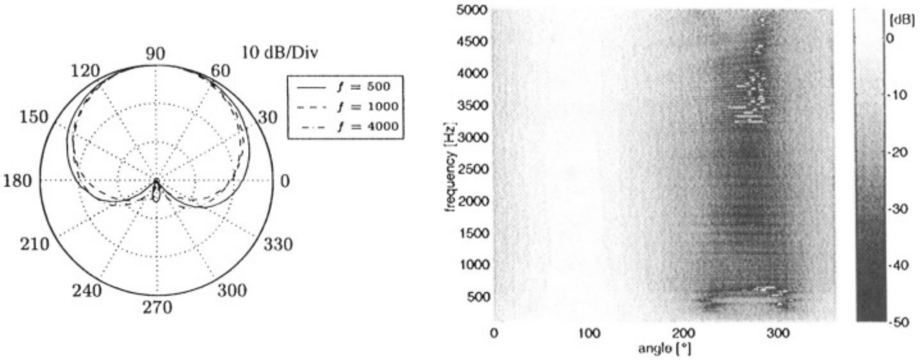


Figure 3.15 **Beam pattern with medium directivity index (DI ≈ 6 dB).**

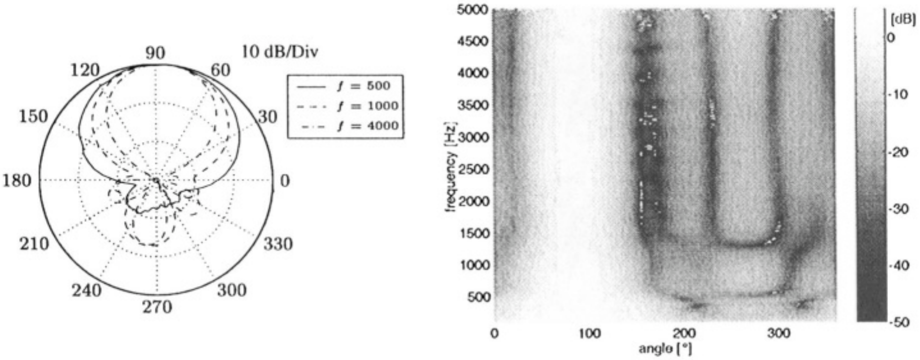


Figure 3.16 **Beam pattern with maximum directivity index (DI ≈ 12 dB).**

modal-beamformer are expressed as the standard spherical harmonics which are well known to be orthonormal. This beamformer architecture yields a computationally efficient implementation for the modal-beamformer. Another advantage of the two-stage beamformer approach is the inherent scalability of the design to any desired modal order. The modal-beamformer is now independent of the number of sensors and the actual geometry of the array (although the geometry of the array does have to meet certain strict requirements). Also, depending on the desired pattern or application, the beamformer has to include only some of the harmonics provided by the decomposer. The spherical array is suitable for a broad range of applications such as directional sound field pick-up, teleconferencing, multi-channel and surround audio, sound-field analysis and synthesis, room acoustic measurement and post recording spatial sound field editing.

References

[1] R. H. DuHamel, "Pattern synthesis for antenna arrays on circular, elliptical and spherical surfaces," Tech. Rep. 16, Electrical Engineering Research Laboratory, University of Illinois, Urbana, 1952.

[2] M. Hoffman, "Conventions for the analysis of spherical arrays," *IEEE Trans. Antennas Propagat.,* vol. 11, pp. 390-393, Jul. 1963.

[3] A. K. Chan, A. Ishimaru, and R. A. Sigelmann, "Equally spaced spherical arrays," *Radio Science,* vol. 3, No. 5, May 1968.

[4] B. Preetham Kumar and G. R. Branner, "The far-field of a spherical array of point dipoles," *IEEE Trans. Antennas Propagat.,* vol. 42, pp. 473-477, Apr. 1994.

[5] P. G. Craven and M. A. Gerzon, "Coincident microphone simulation covering three dimensional space and yielding various directional outputs," US Patent 4,042,779, Jul. 1975.

[6] R. K. Furness, "Ambisonics - an overview," in *Proc. of the AES 8th international conference,* Washington, 1990.

[7] G. W. Elko and A.-T. Nguyen Pong, "A steerable and variable first-order differential microphone array," in *Proc. of IEEE ICASSP,* Munich, 1997.

[8] J. Daniel, *Representation de champs acoustique, application a la transmission et a la reproduction de scene sonores complexes dans un contexte multimedia,* PhD thesis, University Paris 6, 2000.

[9] J. Meyer and G. W. Elko, "A highly scalable spherical microphone array based on an orthonormal decomposition of the soundfield," in *Proc. of IEEE ICASSP,* Orlando, 2002.

[10] T. D. Abhayapala and D. B. Ward, "Theory and design of high order sound field microphones using spherical microphone array," in *Proc. of IEEE ICASSP,* Orlando, 2002.

[11] A. Laborie, R. Bruno and S. Montoya, "A new comprehensive approach of surround sound recording," in *Proc. of the 114th AES Convention*, Amsterdam, 2003.

[12] P. M. Morse and K. U. Ingard, *Theoretical Acoustics,* McGraw-Hill, New York, 1968.

[13] E. G. Williams, *Fourier Acoustics*, Academic Press, San Diego, 1999.

[141 J. J. Bowman, T. B. A. Senior, and P. E. Uslenghi, *Electromagnetic and Acoustic Scattering by Simple Shapes,* Hemisphere Publishing Corporation, New York, 1987

[15] A. C. Ludwig, "Spherical wave theory," in *The handbook of antenna design,* vol. 1, chap. 2.1, Peregrinus, New York, 1983

[16] E. N. Gilbert and S. P. Morgan, "Optimum design of antenna arrays subject to random variations," *Bell Syst. Tech. J.,* 34, pp. 637-663, May 1955.

[17] G. W. Elko, "Superdirectional Microphone Arrays," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., Boston, MA: Kluwer Academic, 2000.

[18] H. Cox, R. M. Zeskind, and T. Kooij, "Practical supergain," *IEEE Trans. ASSP* 34, pp. 393-398, 1986.

[19] G. H. Golub and C. F. van Loan, *Matrix computations,* The Johns Hopkins University Press, Baltimore and New York, 1996.

9. APPENDIX A

Table 3.1 Locations of the sensors of a 32 element spherical array (truncated icosahedron scheme).

sensor No.	φ [°]	ϑ [°]	sensor No.	φ [°]	ϑ [°]
1	180	0	17	252	37.4
2	0	63.4	18	-72	142.6
3	72	63.4	19	216	142.6
4	144	63.4	20	144	142.6
5	216	63.4	21	72	142.6
6	-72	63.4	22	0	142.6
7	36	116.6	23	36	79.2
8	108	116.6	24	72	100.8
9	180	116.6	25	108	79.2
10	252	116.6	26	144	100.8
11	-36	116.6	27	180	79.2
12	0	180	28	216	100.8
13	-36	37.4	29	252	79.2
14	36	37.4	30	-72	100.8
15	108	37.4	31	-36	79.2
16	180	37.4	32	0	100.8

Table 3.2 Locations of the sensors of a 24 element spherical array (*extended* icosahedron scheme).

sensor No.	φ [°]	ϑ [°]	sensor No.	φ [°]	ϑ [°]
1	0	37.4	13	30	100.8
2	60	37.4	14	90	100.8
3	120	37.4	15	150	100.8
4	180	37.4	16	210	100.8
5	240	37.4	17	270	100.8
6	300	37.4	18	330	100.8
7	0	79.2	19	30	142.6
8	60	79.2	20	90	142.6
9	120	79.2	21	150	142.6
10	180	79.2	22	210	142.6
11	240	79.2	23	270	142.6
12	300	79.2	24	330	142.6

This page intentionally left blank

Chapter 4

SUBBAND NOISE REDUCTION METHODS FOR SPEECH ENHANCEMENT

Eric J. Diethorn
Avaya Labs, Avaya
ejd@avaya.com

Abstract Digital noise reduction processing is used in many telecommunications applications to enhance the quality of speech. This investigation focuses on the class of single-channel noise reduction methods employing the technique of short-time spectral modification, a class that includes the popular method of spectral subtraction. The simplicity and relative effectiveness of these subband noise reduction methods has resulted in explosive growth in their use for a variety of speech communications applications. The most commonly used forms of the short-time spectral modification method are discussed, including the Wiener filter, magnitude subtraction, power subtraction, and generalized parametric subtraction. Because of its importance to the subjective performance of any noise reduction method, the subject of real-time signal- and noise-level estimation is also reviewed. A low-complexity noise reduction algorithm is also presented and its implementation is discussed.

Keywords: Noise Reduction, Wiener Filtering, Spectral Subtraction, Short-Time Fourier Analysis, Subband Filter Banks, Implementation

1. INTRODUCTION

Noise enters speech communications systems in many ways. In traditional wire-line telephone calls, one or both parties may be speaking within an environment having high levels of background noise. Calls made from public telephone booths located near roadways, transportation stations, and shopping areas serve as examples. Similarly, cellular, or wireless, telephones permit users to place calls from virtually any location, and it is common for such communications to be degraded by noise of varied origin. In room teleconferencing applications, in which the acoustical characteristics of the environment are generally assumed to be controlled (quiet), it is not uncommon for heating, ventilation and air-conditioning systems to contribute substantial levels of noise. Noise originates not only from acoustical sources, however. Circuit noise, generated electrically within the telephone network, is still prevalent throughout the global telecommunications system.

When present at small or moderate additive levels, noise degrades the subjective quality of speech communications. Listening tests broadly show that people grow less tolerant of, and less attentive to, listening material as the signal-to-noise (SNR) ratio of the material decreases. This phenomenon is known as listener fatigue. When the SNR of speech material is very low, say less than 10 dB, the intelligibility of speech is affected.

Even traditionally low levels of noise can present a problem, especially when multiple speech channels are combined as in conferencing or bridging. In multiparty, or multipoint, teleconferencing, the background noise present at the microphone(s) of each point of the conference combines additively at the network bridge with the noise processes from all other points. The loudspeaker at each location of the conference therefore reproduces the combined sum of the noise processes from all other locations. This problem becomes serious as the number of conferencing points increases. Consider a three-point conference in which the room noise at all locations is stationary and independent with power P. Each loudspeaker receives noise from the other two locations, resulting in a total received noise power of $2P$, or 3 dB greater than that of a two-point conference. With N points, each side receives a total noise power that is $10 \log(N - 1)$ dB greater than P. For example, in a conference with 10 participating locations, the received noise power at each point is about 10 dB greater than that of the two-party case. Because a 10 dB increase in sound power level roughly translates to a doubling of perceived loudness, the noise level perceived by each participant is twice as loud as that of the two-party case. The benefits of noise reduction processing for cases such as this are clearly evident.

A variety of approaches have been proposed to reduce noise for purposes of speech enhancement. Included are: classic (static) Wiener filtering [6]; dynamic comb filtering (see citations in [14]), in which a linear filter is adapted to pass only the harmonic components of voiced speech as derived from the pitch period; dynamic, linear all-pole and pole-zero modeling of speech (see citations in [14]), in which the coefficients of the (noise-free) model are estimated from the noisy speech; short-time spectral modification techniques, in which the magnitude of the short-time Fourier transform is attenuated at frequencies where speech is absent [1]–[5], [11]–[20], [23]–[25]; and hidden Markov modeling [21, 22], a technique also employing time-varying models for speech but where the evolution of model coefficients is governed by transition probabilities associated with model states.

Predominately, speech noise reduction systems are used to improve the subjective quality of speech, lessening the degree to which listener fatigue limits the perceived quality of speech communications. Though some work [14, 24], has shown that intelligibility improvement is possible through digital noise-reduction processing, these results apply to carefully designed listening tests.

All the above noise reduction methods share the property that they operate on a single channel of noisy speech. They are blind techniques, in the sense that only the noise-corrupt speech is known to the algorithm. Thus, in order to enhance the speech-signal-to-noise ratio, the algorithms must form bootstrap estimates of the signal and noise. When multiple channels containing either the same noisy speech source or noise source alone are available, a wide range of spatial acoustic processing technologies applies. Included are adaptive beamformers and adaptive noise cancelers. Adaptive noise cancellation methods are coherent noise reduction processors, exploiting the phase coherency among multiple time series channels to cancel the noise, whereas the noise reduction methods listed above are incoherent processors.

This work focuses solely on the class of single-channel noise reduction methods employing short-time spectral modification techniques. The simplicity and relative effectiveness of these methods has resulted in explosive growth in their use for a variety of speech communications applications. Today, noise reduction processors appear in a variety of commercial products, including cellular telephone handsets; cellular hands-free, in-the-car telephone adjuncts; room teleconferencing systems; in-network speech processors, such as bridges and echo cancelers; in-home telephone appliances, including speakerphones and cordless phones; and hearing aid and protection devices. Frequently, noise reduction processors are commonly used in conjunction with other audio and speech enhancement devices. In room teleconferencing systems, for example, noise reduction is often combined with acoustic echo cancellation and microphone-array processing (beamforming). In summary, the diversity and complexity of modern communications systems present ample opportunity to apply methods of digital noise reduction processing.

The organization of this chapter is as follows. We begin in Section 2 with a brief review of Wiener filtering because of its fundamental relation to all past and modern noise reduction methods that employ spectral modification. Section 3 reviews the technique of short-time Fourier analysis and discusses its use in noise reduction. The short-time Wiener filter is discussed, and a variety of commonly used variations on the Wiener filter are reviewed. Techniques for estimating the signal and noise envelopes are reviewed in Section 4. Last, Section 5 presents a low-complexity implementation of a noise reduction processor for speech enhancement.

2. WIENER FILTERING

Consider the problem of recovering a signal $s(n)$ that is corrupted by additive noise. Let

$$y(n) = s(n) + v(n) \tag{4.1}$$

represent the noisy signal and let the power spectrum of the noise source $v(n)$ be known or, at least, accurately estimated. When $s(n)$ and $v(n)$ areuncorrelated stationary random processes, the power spectrum of the noise-corrupt signal, $P_y(\omega)$, is simply the sum of the power spectrums of the signal and noise:

$$P_y(\omega) = P_s(\omega) + P_v(\omega). \tag{4.2}$$

Under these circumstances the power spectrum of the signal is easily recovered by exploiting (4.2) to subtract the power spectrum of the noise from that of the noisy observation, that is,

$$P_s(\omega) = P_y(\omega) - P_v(\omega). \tag{4.3}$$

Though trivial in concept, this fundamental spectral power subtraction relation forms the basis for the noise reductions methods discussed throughout this chapter.

Of course, (4.3) only provides recovery of the power spectrum of the random process to which sample function $s(n)$ is associated. To estimate $s(n)$ we rely upon classical linear estimation theory. The estimate $\hat{s}(n)$ of $s(n)$ that minimizes the mean-squared error $\|s(n) - \hat{s}(n)\|^2$ is given by [6]

$$\hat{S}_{\mathrm{W}}(\omega) = H_{\mathrm{W}}(\omega)Y(\omega) \tag{4.4}$$

where $\hat{S}_{\mathrm{W}}(\omega)$ is the Fourier transform corresponding to the optimum $\hat{s}(n)$, $Y(\omega)$ is the Fourier transform of $y(n)$, and

$$H_{\mathrm{W}}(\omega) = \frac{P_s(\omega)}{P_s(\omega) + P_v(\omega)} \tag{4.5}$$

is the (noncausal) Wiener filter frequency response function derived by Norbert Wiener many years ago. Thus, the least-mean-square estimate of the signal is acquired simply by applying a frequency dependent gain function to the spectrum of the noisy signal. Note that by using (4.2) and (4.3) in (4.5) and expanding (4.4) we also have

$$\hat{S}_{\mathrm{W}}(\omega) = \left[\frac{P_y(\omega) - P_v(\omega)}{P_y(\omega)}\right] Y(\omega). \tag{4.6}$$

Equation (4.6) illustrates a form of the Wiener recovery method that utilizes a spectral subtraction operation.

To apply Wiener's theory we must make several approximations, which in practice do not limit the usefulness of Wiener's result. When $P_y(\omega)$ and $P_v(\omega)$ are not known they can be estimated from the observed signals. Using the principle of ensemble averaging for stationary signals, the power spectrums of the signal and noise are given by the expected value of the squared-modulous of their respective Fourier transforms. With $E\{\cdot\}$ denoting expectation, we have $P_s(\omega) = E\left\{|S(\omega)|^2\right\}$, $P_v(\omega) = E\left\{|V(\omega)|^2\right\}$ and, consequently, $P_y(\omega) = E\left\{|Y(\omega)|^2\right\}$. Substituting expected values into (4.2) and (4.3), (4.5) gives

$$H_W(\omega) = \frac{E\{|Y(\omega)|^2\} - E\{|V(\omega)|^2\}}{E\{|Y(\omega)|^2\}}. \tag{4.7}$$

When the ensemble averages themselves are unknown we may go one step further. The Fourier transforms of the observed signals can be used as sample estimates of the ensemble averages, leading to

$$H_W(\omega) \cong \frac{|Y(\omega)|^2 - |V(\omega)|^2}{|Y(\omega)|^2} \tag{4.8}$$

as an estimate of the Wiener filter. This form of the Wiener filter serves as the basis for the large majority of spectral-based noise reduction techniques in use today.

3. SPEECH ENHANCEMENT BY SHORT-TIME SPECTRAL MODIFICATION

Wiener filter theory applies to stationary signals and their power spectrums. Speech is, of course, not stationary; its spectral content evolves with time. Further, although in many applications noise source $v(n)$ is accurately modeled as a stationary process, its power spectrum is not known exactly and must be estimated. Under these circumstances Wiener's theory can be applied in a block-processing arrangement using short-time Fourier analysis. In the next several sections we review the short-time Wiener filter method of noise reduction as well as a few of the most commonly used variants.

3.1 SHORT-TIME FOURIER ANALYSIS AND SYNTHESIS

Any time series $x(n)$, stationary or otherwise, can be represented by its short-time Fourier transform (STFT) [5, 8, 9]

$$X(k,m) = \sum_{n=0}^{N-1} h(n)x(m-n)e^{-j2\pi kn/K}, \; k = 0,\ldots,K-1 \tag{4.9}$$

where m is the time index about which the short-time spectrum is computed, k is the discrete frequency index, $h(n)$ is an analysis window, N dictates the duration over which the transform is computed, and K is the number of frequency bins at which the STFT is computed. For stationary signals the magnitude-squared of the STFT provides a sample estimate of the power spectrum of the underlying random process.

A key motivation for using the STFT in speech enhancement applications is that there exist synthesis formulae by which a time series can be exactly reconstructed from its STFT representation. As a result, if the noise can be eliminated from the STFT of $y(n)$, the signal estimate $\hat{s}(n)$ can be recovered through the appropriate synthesis procedure. A STFT analysis and synthesis structure is a type of subband filter bank, and filter bank theory specifies criteria under which perfect reconstruction is possible [5]. Proper synthesis of the time series from its STFT is key to the performance of any noise reduction method. Improper or ad hoc synthesis results in audible artifacts in the reconstructed speech time series, artifacts which reduce the quality of the very material for which enhancement is desired.

In typical noise reduction applications N in (4.9) falls within the range 32–256 for speech time series sampled at 8 kHz. The number of frequency bins, or subband time series channels, K is often within this same range. Analysis windows $h(n)$ for subband filter banks can be designed for particular characteristics [5]. Alternatively, any of the common data windows (e.g., Hamming) can be used, though such traditional windows place constraints on filter bank structure that must be considered by the designer.

Filter bank theory is not discussed further in this chapter, though the example implementation in Section 5 includes a description of a subband filter bank. The reader is directed to [5] for a complete treatment.

3.2 SHORT-TIME WIENER FILTER

The STFT representation can be used to define a short-time Wiener filter. Replacing the Fourier transforms in (4.8) with their corresponding STFTs yields the short-time Wiener filter

$$H_{\mathrm{W}}(k,m) = \frac{|Y(k,m)|^2 - |V(k,m)|^2}{|Y(k,m)|^2}. \tag{4.10}$$

Replacing the Fourier transforms in (4.4) with their corresponding STFT representations and using (4.10) gives

$$\hat{S}_{\mathrm{W}}(k,m) = \frac{|Y(k,m)|^2 - |V(k,m)|^2}{|Y(k,m)|^2} Y(k,m) \tag{4.11}$$

as the estimate of the STFT of the desired signal. As discussed above, the desired full-band speech time series estimate is recovered from $\hat{S}_{\mathrm{W}}(k,m)$ by using the appropriate synthesis procedure associated with the subband filter bank structure being used.

The short-time Wiener filter method of noise reduction was studied in [12, 13,14]. Though the Wiener filter was not the first of the spectral modification method to be investigated for noise reduction, its form is basic to nearly all the noise reduction methods investigated over the last forty years.

In any implementation of (4.11), or any other noise reduction method discussed herein, $Y(k,m)$ is computed at every time index m while $V(k,m)$ is computed only for m for which speech is absent. Otherwise, corruption of the noise envelope estimate occurs. When speech is present, $V(k,m)$ must be estimated from past samples. This subject is discussed further in Section 4.

3.3 POWER SUBTRACTION

An alternative estimate of $S(k,m)$ is arrived at by departing from Wiener's theory. $S(\omega)$ can be represented in terms of its magnitude and phase components, namely,

$$S(\omega) = |S(\omega)|e^{j\phi_s(\omega)}. \tag{4.12}$$

Thus, $S(\omega)$ can be estimated if estimates of its magnitude and phase can be found. Consider, first, stationary $s(n)$ and $v(n)$. As above, if the sample functions $|S(\omega)|^2$ and $|V(\omega)|^2$ are used in place of the power spectrums $P_s(\omega)$ and $P_v(\omega)$, (4.3) becomes

$$|S(\omega)|^2 = |Y(\omega)|^2 - |V(\omega)|^2. \tag{4.13}$$

The square root of (4.13) therefore provides an estimate of the signal's magnitude spectrum. Concerning the signal's phase, if the signal-to-noise ratio of the noisy signal is reasonably high, the phase of the noisy signal, $\phi_y(\omega)$, can be used in place of $\phi_s(\omega)$. Using these magnitude and phase estimates (4.12) yields

$$\hat{S}_{\mathrm{PS}}(\omega) = \sqrt{|Y(\omega)|^2 - |V(\omega)|^2}\; e^{j\phi_y(\omega)} \tag{4.14}$$

as the spectrum estimate of the desired signal. Using STFT quantities in place of the power spectrums in (4.14) yields

$$\hat{S}_{\mathrm{PS}}(k,m) = \sqrt{|Y(k,m)|^2 - |V(k,m)|^2}\; e^{j\phi_y(k,m)} \tag{4.15}$$

as the short-time spectrum estimate. The form in (4.14) and (4.15) is referred to as the power subtraction method of noise reduction. Power subtraction was studied in [4, 12, 13, 14, 17].

Note that (4.14) provides a consistent estimate of the magnitude-squared of $S(\omega)$. Assuming the signal and noise are uncorrelated, we have

$$\begin{aligned} E\{|\hat{S}_{\mathrm{PS}}(\omega)|^2\} &= E\{|Y(\omega)|^2\} - E\{|V(\omega)|^2\} \\ &= E\{|S(\omega)|^2\} \qquad (4.16) \\ &= P_s(\omega). \end{aligned}$$

Like the Wiener filter, the power subtraction method also can be shown to be optimal from within the estimation theoretic framework, albeit from a different formulation of the problem. Let $S(k,m)$ and $V(k,m)$ be realizations of independent, stationary Gaussian random processes. Also, without loss of generality, assume $S(k,m)$ and $V(k,m)$ are real. If σ_S^2 and σ_V^2 are the variances of the signal and noise STFTs, the probability density function of the observation $Y(k,m)$ given the known signal and noise variances is given by [7, 12]

$$p\left(Y(k,m) \mid \sigma_S^2, \sigma_V^2\right) = \frac{1}{\pi\left(\sigma_S^2 + \sigma_V^2\right)} e^{-\frac{Y^2(k,m)}{\sigma_S^2+\sigma_V^2}} . \tag{4.17}$$

The maximum likelihood estimate of the signal variance is the estimate $\hat{\sigma}_S^2$ that maximizes the likelihood of the noisy observation occurring. Maximizing (4.17) with respect to the signal variance gives

$$\hat{\sigma}_s^2 = Y^2(k,m) - \sigma_V^2 \tag{4.18}$$

as the best estimate of the signal variance. This estimate suggests the power subtraction estimator (4.15) when variances are replaced by sample approximations, that is, by the corresponding magnitude-squared STFT quantities. Thus, the power subtraction estimator results from the optimum maximum likelihood signal-variance estimator. In comparison, the Wiener estimator results from the optimum minimum mean-squared error estimate of the signal spectrum, or equivalently, the signal time series.

McAulay and Mulpass [13] show that the time domain dual of (4.17), in which $y(n)$ replaces $Y(k,m)$ and the variances are of the signal and noise time series, also yields the power subtraction form when maximized with respect to the unknown signal variance.

3.4 MAGNITUDE SUBTRACTION

Yet another estimate of $S(k,m)$ is suggested by the magnitude-only form of (4.3). Consider

$$|\hat{S}(k,m)| = |Y(k,m)| - |V(k,m)| \tag{4.19}$$

as an estimate of the magnitude $S(k,m)$. Using (4.19) in (4.12), and appending the phase of the noisy signal as done in Section 3.3, gives

$$\hat{S}_{\mathrm{MS}}(k,m) = \left[|Y(k,m)| - |V(k,m)|\right] e^{j\phi_y(k,m)} \tag{4.20}$$

as the short-time spectrum estimate of the signal. The form (4.20) is referred to as the magnitude subtraction method of noise reduction. This method of noise reduction was popularized by Boll [11], but was suggested by Weiss et al. [15] and even earlier by Schroeder [1, 2, 4].

3.5 PARAMETRIC WIENER FILTERING

The Wiener, power subtraction and magnitude subtraction schemes are closely related. Several authors have exploited this fact to define a general class of noise reduction methods. To see this, first note that the power subtraction estimate (4.15) can be rewritten as

$$\begin{aligned}\hat{S}_{\mathrm{PS}}(k,m) &= \sqrt{|Y(k,m)|^2 - |V(k,m)|^2}\, e^{j\phi_y(k,m)} \\ &= H_{\mathrm{PS}}(k,m)Y(k,m),\end{aligned} \tag{4.21}$$

where

$$H_{\mathrm{PS}}(k,m) = \left[\frac{|Y(k,m)|^2 - |V(k,m)|^2}{|Y(k,m)|^2}\right]^{1/2}. \tag{4.22}$$

Similarly, the magnitude subtraction estimate (4.20) can be rewritten as

$$\begin{aligned}\hat{S}_{\mathrm{MS}}(k,m) &= \big[|Y(k,m)| - |V(k,m)|\big] e^{j\phi_y(k,m)} \qquad (4.23)\\ &= H_{\mathrm{MS}}(k,m)Y(k,m),\end{aligned}$$

where

$$H_{\mathrm{MS}}(k,m) = \frac{|Y(k,m)| - |V(k,m)|}{|Y(k,m)|}. \tag{4.24}$$

Both forms result from the generalized estimate

$$\hat{S}_{\mathrm{G}}(k,m) = H_{\mathrm{G}}(k,m)Y(k,m), \tag{4.25}$$

where

$$H_{\mathrm{G}}(k,m) = \left[1 - \left(\frac{|V(k,m)|}{|Y(k,m)|}\right)^{\gamma}\right]^{\beta}. \tag{4.26}$$

This has been referred to as parametric Wiener filtering [14] or parametric spectral subtraction [17]. Power subtraction results from using $(\gamma,\beta) = (2,1/2)$, magnitude subtraction from $(\gamma,\beta) = (1,1)$, and the Wiener estimate from $(\gamma,\beta) = (2,1)$.

For general (γ,β) the parametric form (4.25) has not been demonstrated to satisfy optimality criteria, though this fact does not in any way diminish the usefulness of the generalized form for choosing noise reduction gain formulae.

Figure 4.1 shows a plot of the Wiener (4.10), power subtraction (4.22) and magnitude subtraction (4.24) noise reduction gain functions as a function of a *priori* SNR. The *a priori* SNR is the ratio σ_s^2/σ_v^2. Also, for purposes of evaluating each gain function, the magnitude-squared sample spectrums have been replaced by their respective variances. The curves show the attenuation of each gain function as input SNR decreases. Note that the Wiener- and power-subtraction-gain functions provide an attenuation of no greater than 6 dB at an

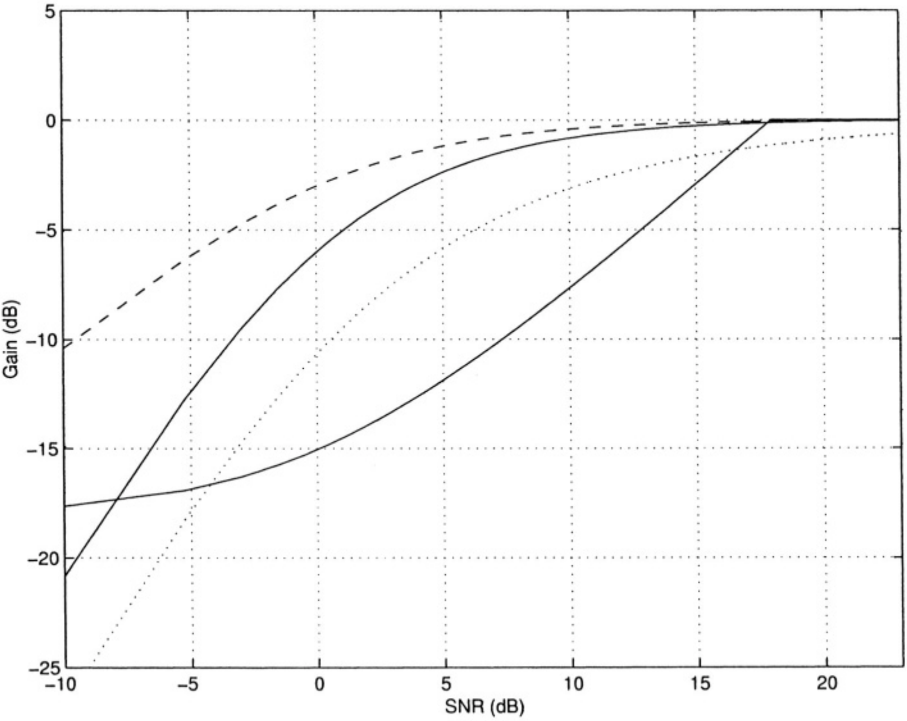


Figure 4.1 Gain functions for the Wiener (upper-most solid), spectral power subtraction (dash), spectral magnitude subtraction (dot), and *a posteriori* SNR voice activity detection (lower-most solid) methods of noise reduction as a function of *a priori* input signal-to-noise ratio.

input SNR of 0 dB. That is, in any subband k in which the signal and noise powers are equal, the contribution of that subband to the reconstructed speech time series is no less than half its input amplitude. A fourth gain curve appearing in the figure, namely, that of a voice-activity-detection-based method, will be discussed in Section 5.

Gain curves such as Fig. 4.1 provide some insight into the nature of a given noise reduction algorithm, but provide little indication of the subjective speech quality of the resulting noise-reduced signal. More important to the subjective quality of a noise reduction algorithm is the manner in which the speech and noise envelopes are estimated; this subject is discussed in Section 4.

3.6 REVIEW AND DISCUSSION

3.6.1 Schroeder's Noise Reduction Device. The first use of spectral gain modification methods in speech noise reduction is described in a little-known U.S. patent issued in 1965 to M. R. Schroeder [1]–[4], who at the time was

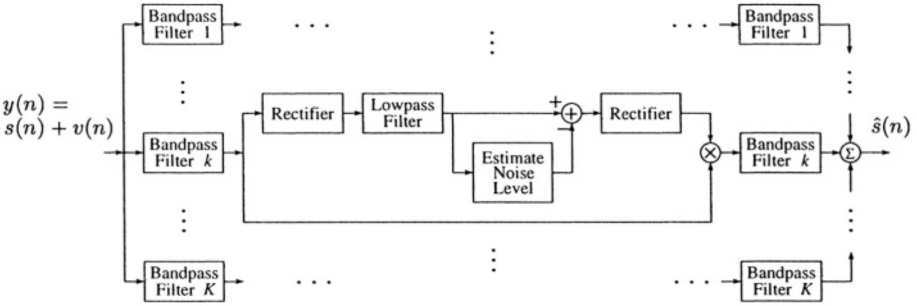


Figure 4.2 Schroeder's noise reduction system. After M. R. Schroeder [1, 2].

working for AT&T Bell Laboratories. A block diagram of Schroeder's noise reduction system is shown in Fig. 4.2. This diagram is modified from its original form for this discussion, and incorporates elements presented by Schroeder in a related 1968 patent [2] and also by Schroeder's colleagues in a subsequent published work [4].

Schroeder's system was a purely analog implementation of spectral magnitude subtraction. As shown in the figure, a bank of bandpass filters separates the noisy signal into *K* different frequency bands. The bandwidth of each filter is about 300 Hz. Ten individual filters therefore cover the 300 to 3300 Hz range necessary for telephony grade speech applications. The noise-reduction processing performed in each band is identical. First, the output of each filter bank is rectified and averaged using a low-pass filter to produce a short-time estimate of the noisy speech envelope for the band. The lowpass filter has a cutoff of between 0 and 10 Hz. The noisy speech envelope is then subtracted from an estimate of the noise-only envelope. To estimate the noise, the noise level estimator uses a series of resistors, capacitors and diodes to produce a running estimate of the minima of the noisy speech envelope. The decay time of this noise estimator is instantaneous while the rise time is very large, on the order of seconds. Between speech utterances the noisy speech envelope contains only noise, and the noise level estimator quickly decays to meet the level of the noise. During utterances the noise level estimate changes very little. Thus, the output of the subtraction block is an estimate of the noise-free signal envelope for the band, or $|\hat{S}(k,m)|$ in the current notation. A second rectification is performed on the output to accommodate negative results from the difference node (negative estimates are simply set to zero). Finally, the noise-free signal envelope is used as a multiplier with the unmodified output of the bandpass filter for the band, and the result is summed with the results from all bands to form the reconstructed full-band time series $\hat{s}(n)$.

It is interesting to note that Schroeder's implementation was a purely analog one, employing bandpass filters and rectification and averaging circuitry. Other aspects of the design preceded presentations by later authors. By rectifying the output of the signal envelope estimator, Schroeder's system could correct for negative estimates.

3.6.2 Literature Review. Following Schroeder's work, it was not until the mid-1970s that interest in noise reduction systems grew, presumably because of the availability of digital computers and analog processors that could be controlled by digital decision logic. With the unification of digital multirate processing theory in the late 1970s and 1980s came the realization that Wiener-filter processing, among other operations, could be accomplished efficiently using digital subband architectures [5, 8, 9].

Digital noise reduction processing for speech enhancement was popularized by the technique of spectral subtraction. This renewed interest appears to have been sparked in a 1974 paper by Weiss, Aschkenasy, and Parsons [15]. Their paper describes a "spectrum shaping" method that used amplitude clipping, or gating, in filter banks to remove low-level excitation, presumably noise. A few years later, Boll [11], in an often-sited reference, was apparently the first to reintroduce the spectral subtraction method that Schroeder had identified nearly 20 years earlier. Boll was perhaps the first to cast the magnitude subtraction method in the framework of digital short-time Fourier analysis, which had earlier been under development by, among others, Allen [9] and Protnoff [8]. Shortly after, McAulay and Malpass [13] presented one of the first treatments of the spectral subtraction method from within a framework of optimal estimation, which included Wiener filter theory. They described a class of spectral subtraction estimators, including power subtraction and magnitude subtraction, from within an estimation theoretic framework [7]. Coincident with [13], Lim and Oppenheim [14] presented one of the first comprehensive treatments of methods of speech enhancement and noise reduction. The spectral subtraction methods are discussed, also within a framework of optimal estimation, and are compared to other methods of speech enhancement.

Lim and Oppenheim recognize Weiss, Aschkenasy, and Parsons as originators of the spectral subtraction technique, as Schroeder's work was likely unknown to them at the time. Also, in 1980, Sondhi, Schmidt, and Rabiner [4] published results from a series of implementation studies that grew from Schroeder's work in the 1960's. This work was the first published reference of Schroeder's work, other than the patents [1, 2], but likely was not widely known itself because it was published in the Bell System Technical Journal.

Yet another spectral noise reduction method has been proposed by Ephraim and Malah [12]. They derive a related spectral noise reduction method based on optimum short-time Fourier amplitude estimation. Differing from a variance estimator, that is, power subtraction, the amplitude estimator is optimum in the sense of providing the best minimum mean-squared error estimate of the spectral amplitude. The Fourier amplitude estimator converges to the Wiener estimator at high input signal-to-noise ratios.

Recently, noise reduction processing incorporating psychoacoustic perceptual models has been proposed. Tsoukalas and Mourjopoulos [24] present a spectral gain modification technique that uses perceptual models to suppress only those components of the noise that are above audibility thresholds. These thresholds are dynamic, changing with the changing spectral character of the speech itself. A reported 40% gain in intelligibility can be achieved when precise information about the noise power level is known.

3.6.3 Musical Noise. Much of the work in noise reduction in the last 20 years has been directed toward implementation issues associated with spectral subtraction methods, particularly in understanding and eliminating a host of processing artifacts commonly referred to as musical noise. Musical noise is a processing artifact that has plagued all spectral modification methods. This artifact is perceived by many as the sound made by an ensemble of low-amplitude tonal components, the frequencies of which are changing rapidly over time. The amplitude of these components is usually small, on the order of the noise power itself. First, because the STFTs in (4.26) are computed over short intervals of time, and because the noise envelope estimate is made only during periods of silence while the noisy signal envelope estimate is always made, the difference in (4.26) can actually be negative. In such cases the common approach is set the difference to zero or to invert the sign of the difference and use the result. Such harsh action creates sudden discontinuities in the trajectories of spectral amplitudes, and this induces the artifact. Second, artifacts are induced by improper synthesis of the full-band time series. Ad hoc "FFT processing" results in filter banks that do not possess the quality of perfect reconstruction and, moreover, cause aliasing of the subband time series in both frequency and time.

The works of a variety of authors have shown that artifacts such as musical noise can be nearly eliminated by taking proper action in chiefly three aspects of the processing. First, careful design of the filter bank is required. Second, the proper use of time-averaging techniques, in conjunction with appropriate decision criteria, is necessary to produce stable estimates and obviate the need for rectification following spectral subtraction. Third, the complementary technique of augmenting the traditional gain function with a soft-decision, voice-activity-detection (VAD) statistic has proven extremely successful. This later technique supplants the traditional gain-based noise reduction form in (4.25) with

$$\hat{S}_{\mathrm{G}}(k,m) = H_{\mathrm{G}}(k,m)P\big(H_1 \mid Y(k,m)\big)Y(k,m), \tag{4.27}$$

where H_1 denotes the hypothesis that the signal is present in the observation and $P\big(H_1 \mid Y(k,m)\big)$ is the probability that the signal is present conditioned on the observation. $P\big(H_1 \mid Y(k,m)\big)$ acts as a gain function itself. At very low input SNRs, $P(H_1 \mid Y(k,m))$ further suppresses fluctuations in the traditional gain function, fluctuations caused by the statistical volatility of the signal and noise envelope estimates.

Boll [11] was one of the first to augment a traditional gain function (magnitude subtraction) with a VAD. Boll's method integrates the estimated *a priori* SNR[1] across all frequency bins and uses the resulting sealer in a binary VAD which, if satisfied, applies a fixed additional attenuation to the right-hand side of (4.25). McAulay and Malpass [13] derived a "soft-decision" VAD from within the detection theoretic framework [7]. Unlike a binary VAD, a soft VAD varies continuously within (0,1) as a function of $Y(k,m)$. Later, Ephraim and Malah [12] incorporated a measure of the "signal presence uncertainty" into their Fourier amplitude estimator. Clearly, $H_G(\cdot)$ and $P(\cdot)$ in (4.27) can be combined into a single gain function; this idea is pursued in the implementation presented in Section 5.

3.6.4 A Word About Phase Augmentation. It is often noted that the practice of phase augmentation – that is, using the phase of the noisy signal as an estimate of the signal's phase – is acceptable because the ear is relatively insensitive to phase corruption. More is known, however. Within the context of noise reduction, Vary [20] has shown that as long as the SNR is at least 6 dB in any subband k for which the gain function is near unity, the resulting distortion is generally imperceptible. In other words, because the noisy phase contributes to $\hat{S}(k,m)$ only at those k for which the input SNR is positive, the phase of $\hat{S}(k,m)$ itself is essentially noise-free. Additionally, Ephraim and Malah [12] show that the noisy phase is a good choice because it has the property of not corrupting the envelope of the optimum short-time Fourier amplitude estimator that forms the basis of their method.

4. AVERAGING TECHNIQUES FOR ENVELOPE ESTIMATION

Proper estimation of both the noisy signal envelope and noise envelope is paramount to the performance of any noise reduction technique. Improper estimation of either envelope will result in an unacceptably high level of audible processing artifacts, such as musical noise. Up to this point, instantaneous envelope quantities have been used in the gain formulae of the noise reduction methods discussed. To combat musical noise, however, averaged, or smoothed, envelopes are used. In the next few sections a variety of commonly used time averaging techniques are reviewed and discussed in the context of noise reduction.

4.1 MOVING AVERAGE

One of the simplest ways to improve the stability of the noise estimate is to replace it with an arithmetic average computed over its recent past. For each time index m, $|V(k,m)|$ is replace by a smoothed version $\overline{V}(k,m)$ given by

$$\overline{V}(k,m) = \frac{1}{M} \sum_{l=0}^{M-1} |V(k,m-l)|, \tag{4.28}$$

where *M* is the number of samples used in the average. The over-bar notation in (4.28) shall be used throughout to denote any averaged magnitude quantity, regardless of the averaging technique actually used. In the computation of (4.28) the reader should note that $V(k,m)$ is simply $Y(k,m)$ under the assumption that speech is absent.

Because of statistical variation in the noisy signal, it is also fortuitous to use a smoothed version, $\overline{Y}(k,m)$, in place of $|Y(k,m)|$ in the gain formulae. Though the amount of smoothing necessary depends upon several aspects of the implementation, such as the subband filter bank structure, $\overline{Y}(k,m)$ is generally smoothed much less than $\overline{V}(k,m)$. This is because variations in $\overline{Y}(k,m)$ and $\overline{V}(k,m)$ induce different artifacts in the signal estimate $\hat{S}_{\mathrm{G}}(k,m)$. Referring to (4.26), positive fluctuations in $|V(k,m)|$ can cause the difference in (4.26) to be negative, requiring rectification. Consequently, it is beneficial to average the noise magnitude over long intervals (assuming stationary noise). Similarly, positive fluctuations in $|Y(k,m)|$ resulting from the statistical variability of the noise component reduce the effectiveness of noise reduction because $H_{\mathrm{G}}(k,m)$ is larger for larger $|Y(k,m)|$. Thus, some smoothing of the noisy speech envelope is also beneficial. Excessive smoothing of the noisy speech envelope, however, degrades the speech quality of the signal estimate because $S(k,m)$ is not stationary. Excessive smoothing of $|Y(k,m)|$, and therefore $H_{\mathrm{G}}(k,m)$, disperses $H_{\mathrm{G}}(k,m)$ to the point that it is no longer well matched to the speech component of the noisy observation $Y(k,m)$.

Early on, Boll [11] described the use of arithmetic averaging to reduce the presence of artifacts. For the STFT filter bank implementation used, Boll applied the same sized average to both the noise and noisy speech envelopes (about 38 ms). McAulay and Malpass [13] also discuss arithmetic averaging.

4.2 SINGLE-POLE RECURSION

The arithmetic average requires an *M*-length history of the data. Further, each sample in the average receives the same weight, although it is trivial to include a tapered weighting window in (4.28) if desired. An alternative to arithmetic averaging is recursive averaging. Using a single-pole recursive average the noise envelope estimate becomes

$$\overline{V}(k,m) = \alpha\overline{V}(k,m-1) + (1-\alpha)|V(k,m)|, \tag{4.29}$$

where α, $0 < \alpha < 1$, is the coefficient of smoothing. Equation (4.29) defines a first-order lowpass filter and so the variance of $\overline{V}(k,m)$ is less than the variance of $|V(k,m)|$ itself.

Recursive averaging has been by far the most popular method of averaging used in the spectral noise reduction methods. This is due to its simplicity and efficiency, requiring only a single memory location for state variable storage. Also, because the impulse response corresponding to (29) decays as $\alpha^n, n \geq 0$, the recursive average weights the recent past more heavily than the distant past. This characteristic has been found beneficial to noise reduction processing. Indeed, the first investigations into the use of averaging techniques employed recursive averaging. Schroeder's method (Fig. 4.2) incorporates an analog version of (4.29) in the signal path common to both the noise and noisy signal envelope estimators. Sondhi et al. [4] experimented with variations on the recursive average for both the power subtraction and magnitude subtraction methods. For computing $\overline{V}(k,m)$, cutoff frequencies of between 10 and 30 Hz, or greater, were found to be effective [4]. For estimating $\overline{Y}(k,m)$, cutoff frequencies of between 1 and 10 Hz were sufficient. Other proponents of the recursive average include McAulay and Malpass [13], Ephraim and Malah [12], and Cappé [16].

4.3 TWO-SIDED SINGLE-POLE RECURSION

An alternative to the classic single-pole recursive filter involves choosing α in (4.29) based upon the magnitude of $|V(k,m)|$ relative to $\overline{V}(k,m-1)$. Consider the so-called two-sided single-pole recursion in which α in (4.29) is given by

$$\alpha = \begin{cases} \alpha_{\text{a}}, & \text{if } |V(k,m)| \geq \overline{V}(k,m-1) \\ \alpha_{\text{d}}, & \text{if } |V(k,m)| < \overline{V}(k,m-1) \end{cases}, \tag{4.30}$$

where α_{a} is the "attack" coefficient and α_{d} is the "decay" coefficient. The two-sided recursive average employs two different filter response times, depending on whether the input is increasing or decreasing in magnitude relative to the current average. This property can be advantageous. Consider, first, the computation of $\overline{V}(k,m)$. Although it is desirable to update $V(k,m)$ only when speech is absent, it is not always possible to determine when speech is present and when it is not. If speech or other transient phenomena are present in $Y(k,m)$ and (4.29) is updated, $\overline{V}(k,m)$ will become corrupt. This problem can be reduced by choosing $\alpha_{\text{a}} > \alpha_{\text{d}}$. In this case increases in $|V(k,m)|$ change $\overline{V}(k,m)$ much less than decreases in $|V(k,m)|$, and therefore $\overline{V}(k,m)$ is less perturbed by transient phenomenon that are not components of the stationary noise.

The two-sided single-pole recursion can also be used to update $\overline{Y}(k,m)$. For this purpose it is common to choose $\alpha_{\mathrm{a}} < \alpha_{\mathrm{d}}$, in which case $\overline{Y}(k,m)$ is more responsive to the sudden onset of speech energy than to the end of an utterance when speech energy decays. This characteristic improves the response of $H_G(k,m)$ in (4.26) to the onset of speech.

Etter and Moschytz [17] and Diethorn [19] used the two-sided single-pole recursion in the context of noise reduction; it is also used in the implementation discussed in Section 5. The technique has it origins in speakerphone technology, where it is used for voice activity detection; for example, see [26].

As a variation on (4.29)–(4.30), Etter and Moschytz [17] also proposed using

$$\overline{V}(k,m) = \begin{cases} \alpha_{\mathrm{a}}\overline{V}(k,m-1), & \text{if } a_{\mathrm{a}}\overline{V}(k,m-1) < |V(k,m)| \\ \alpha_{\mathrm{d}}\overline{V}(k,m-1), & \text{if } a_{\mathrm{d}}\overline{V}(k,m-1) > |V(k,m)| \\ |V(k,m)|, & \text{otherwise} \end{cases}, \quad (4.31)$$

where $\alpha_{\mathrm{a}} > 1$ and $0 < \alpha_{\mathrm{d}} < 1$. In subjective listening tests, this so-called two-slope limitation filter reportedly performs better than (4.29)–(4.30) for some material [17].

4.4 NONLINEAR DATA PROCESSING

To improve the stability of the noise estimate further, Sondhi et al. [4] also experimented with a scheme to post-process the noise envelope $\overline{V}(k,m)$ based upon a short-term histogram of its past values. This early rank ordering technique provided a means to prune wild points from the noise envelope estimate. Median filtering and other rank-order statistical filtering can be used to post-process $\overline{V}(k,m)$ and $\overline{Y}(k,m)$ following any of the averaging techniques described above; see [10] for an early reference on such methods. More recently, Plante et al. [25] have described a noise reduction method using reassignment methods to replace envelope estimates that are deemed erroneous. In general, nonlinear data processing techniques can provide improved noise reduction performance, although the behavior of such methods is sometimes difficult to analyze analytically.

5. EXAMPLE IMPLEMENTATION

Noise reduction systems need not be complicated to produce acceptable results, as the implementation described in this section demonstrates. The method described is a variation on that first presented in [19].

Figure 4.3 shows a signal-flow diagram of the noise reduction system. The algorithm consists of four key processes: subband analysis, envelope estima-

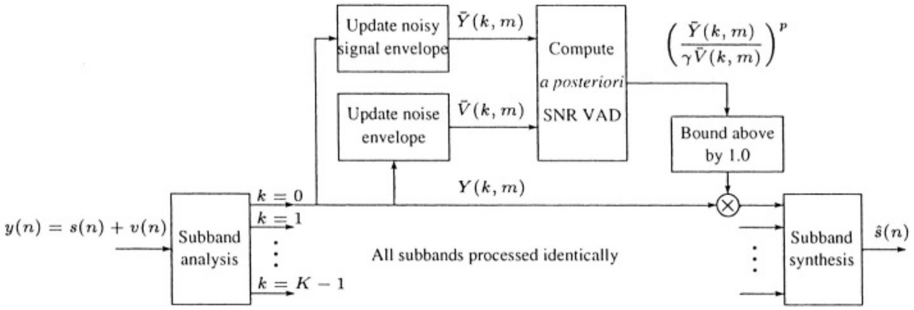


Figure 4.3 Noise reduction system based on *a posteriori* SNR voice activity detection.

tion, gain computation, and subband synthesis. Each of these components is described below.

5.1 SUBBAND FILTER BANK ARCHITECTURE

The subband architecture implements a perfect reconstruction filter bank using the uniform discrete Fourier transform (DFT) filter bank method [5]. This filter bank is one of a sub-class of so-called polyphase filter banks, but is somewhat simpler in computational structure.

The subband filter bank implements an overlap-add process. At the start of each processing epoch, a block of L new time-series samples is shifted into an N-sample shift register. Here, $L = 16$ and $N = 64$. The shift-register data are multiplied by a length-N analysis window (the filter bank's prototype FIR filter) and transformed via an N-point DFT. Each frequency bin output from the DFT represents one new complex time-series sample for the subband frequency range corresponding to that bin. The subband sampling rate is equal to the full-band sampling rate divided by L. The bandwidth of each subband is the ratio of the full-band sampling rate to N. Thus, for 8 kHz sampling, the subband sampling rate and bandwidth are, respectively, 500 Hz and 125 Hz (indicative of an oversampled-by-4 filter bank architecture). Following subband analysis, the vector of subband time series is presented to the envelope estimators. Next, the noise reduction gain is computed. To reconstruct the noise-reduced full-band time series, the subband synthesizer first transforms the gain-modified vector of subband time series using an inverse DFT. The synthesis window (same as analysis window) is applied, and the result is overlapped with, and added to, the contents of an N-sample output accumulator. Last, a block of L processed samples is produced at the output of the synthesizer.

The prototype analysis and synthesis window, input-output block size L and transform block size N are chosen to maintain these properties of the filter bank:

- no time-domain aliasing at the subband level,
- no frequency domain aliasing at the subband level, and
- perfect reconstruction (unity transfer function) when analysis is followed directly by synthesis (no intermediate processing).

5.2 A-POSTERIORI-SNR VOICE ACTIVITY DETECTOR

The gain function of the noise reduction algorithm is based on the idea of a composite gain function and soft-acting voice activity detector as discussed in Section 3.6.3.

5.2.1 Envelope Computations. As shown in Fig. 4.3, the time series output from each subband k of the analysis filter bank is used to update estimates of the noisy speech and noise-only envelopes, respectively, $\overline{Y}(k,m)$ and $\overline{V}(k,m)$. These estimates are generated using the two-sided single-pole recursion described in Section 4.3. Specifically,

$$\overline{Y}(k,m) = \beta\overline{Y}(k,m-1) + (1-\beta)|Y(k,m)|, \tag{4.32}$$

and

$$\overline{V}(k,m) = \alpha\overline{V}(k,m-1) + (1-\alpha)|Y(k,m)|, \tag{4.33}$$

where β takes on attack and decay constants of about 1 ms and 10 ms, respectively, and α takes on attack and decay time constants of about about 4 sec. and 1 ms. Note that, in comparison with (4.29), (4.33) uses $Y(k,m)$ in place of $V(k,m)$. This substitution is possible because the long attack time (4 sec.) of the noise envelope estimate is used in place of the logic that would otherwise be needed to discern the speech/no-speech condition. This approach further simplifies the implementation.

5.2.2 Gain Computation. The envelope estimates are used to compute a gain function that incorporates a type of voice activity likelihood function. This function consists of the *a posteriori* SNR normalized by the threshold of speech activity detection, γ. Specifically, the noise reduction formula is

$$\hat{S}(k,m) = H(k,m)Y(k,m), \tag{4.34}$$

where gain function $H(k,m)$ is given by

$$H(k,m) = \min\left[1,\ \left(\frac{\overline{Y}(k,m)}{\gamma\overline{V}(k,m)}\right)^{p}\right]. \tag{4.35}$$

Threshold γ specifies the *a posteriori* SNR level at which the certainty of speech is declared and p, a positive integer, is the gain expansion factor. Typical values for the detection threshold fall in the range $5 \leq \gamma \leq 20$, though the (subjectively) best value depends on the characteristics of the filter bank architecture and the time constants used to compute the envelope estimates, among other things. The expansion factor p controls the rate of decay of the gain function for *a posteriori* SNRs below unity. With $p = 1$, for example, the gain decays linearly with *a posteriori* SNR. Factor p also governs the amount of noise reduction possible by controlling the lower bound of (4.35); larger p results in a smaller lower bound. The $\min(\cdot)$ operator insures the gain reaches a value no greater than unity.

Looking at (4.35), subband time series whose *a posteriori* SNR exceeds the speech detection threshold are passed to the synthesis bank with unity gain. Subband time series whose *a posteriori* SNR is less than the threshold are passed to the synthesis bank with a gain that is proportional to the SNR raised to the power p.

Note in particular that (4.35) does not involve a spectral subtraction operation. This has the benefit of circumventing the problem of a negative argument, as occurs with the parametric form in (4.26). A disadvantage of (4.35) is that the gain function, and therefore noise reduction level, is bounded below by the reciprocal of the detection threshold. That is, as the *a priori* SNR goes to zero we have (for $p = 1$)

$$\begin{aligned} \frac{|Y(k,m)|}{\gamma|V(k,m)|} &= \frac{|S(k,m) + V(k,m)|}{\gamma|V(k,m)|} \\ &\approx \frac{|V(k,m)|}{\gamma|V(k,m)|} \\ &= \frac{1}{\gamma}. \end{aligned} \tag{4.36}$$

For example, with $\gamma = 10$ the system provides no more than 20 dB of noise reduction.

A variation on the above technique incorporates for each subband k both the per-band, or narrowband, normalized *a posteriori* SNR and a k-wise arithmetic average of the *a posteriori* SNRs from neighboring bands. This narrowband-broadband hybrid gain function can provide improved noise reduction performance for wideband speech utterances, such as fricatives. The reader is referred to [19] for more information.

5.3 EXAMPLE

The time series and spectrogram data in Figs. 4.4, 4.5, and 4.6 show the results of processing a noisy speech sample using the subband noise reduction method presented above. For this example $\gamma = 8$ and expansion factor $p = 1$, resulting in a minimum gain in (4.35) of –18 dB. The lower-most solid line in Fig. 4.1 shows the gain function (4.35) in comparison with the Wiener, magnitude subtraction and power subtraction gain functions.

The upper trace in Fig. 4.4 shows a segment of raw (unprocessed) time series for a series of short utterances (digit counting) recorded in an automobile traveling at highway speeds. The speech was recorded from the microphone channel of a wireless-phone handset and later digitized at an 8 kHz sampling rate. The lower trace in Fig. 4.4 shows the corresponding noise-reduced time series produced by the noise reduction algorithm. Figure 4.5 shows spectrograms corresponding to the time series in Fig. 4.4. The spectrograms show, at least visibly, that the noise reduction method introduces no noticeable distortion. Figure 4.6 shows the averaged power-spectral density of the background noise for the raw and noise-reduced time series. These power spectral densities were computed from the time series in Fig. 4.4 over the interval $10s - 12s$. As can be seen, the noise floor of the processed time series is about 18 dB below that of the raw time series uniformly across the speech band.

6. CONCLUSION

The subject of noise reduction for speech enhancement is a mature one with a 40-year history in the field of telecommunication. The majority of research has focused on the class of noise reduction methods incorporating the technique of short-time spectral modification. These methods are based upon subband filter bank processing architectures, are relatively simple to implement and can provide significant gains to the subjective quality of noisy speech. The earliest of these methods was developed in 1960 by researchers at Bell Laboratories.

Noise reduction processing has its roots in classical Wiener filter theory. Reviewed in this chapter were the most commonly used noise reduction formulations, including the short-time Wiener filter, spectral magnitude subtraction, spectral power subtraction, and the generalized parametric Wiener filter. When implemented digitally, these methods frequently suffer from the presence of processing artifacts, a phenomenon known as musical noise. The origins of musical noise were reviewed, as were approaches to combating the problem. The subject of speech envelope estimation was presented in detail and several averaging techniques for computing envelope estimates were reviewed. A low-complexity noise reduction algorithm was presented and demonstrated by example.

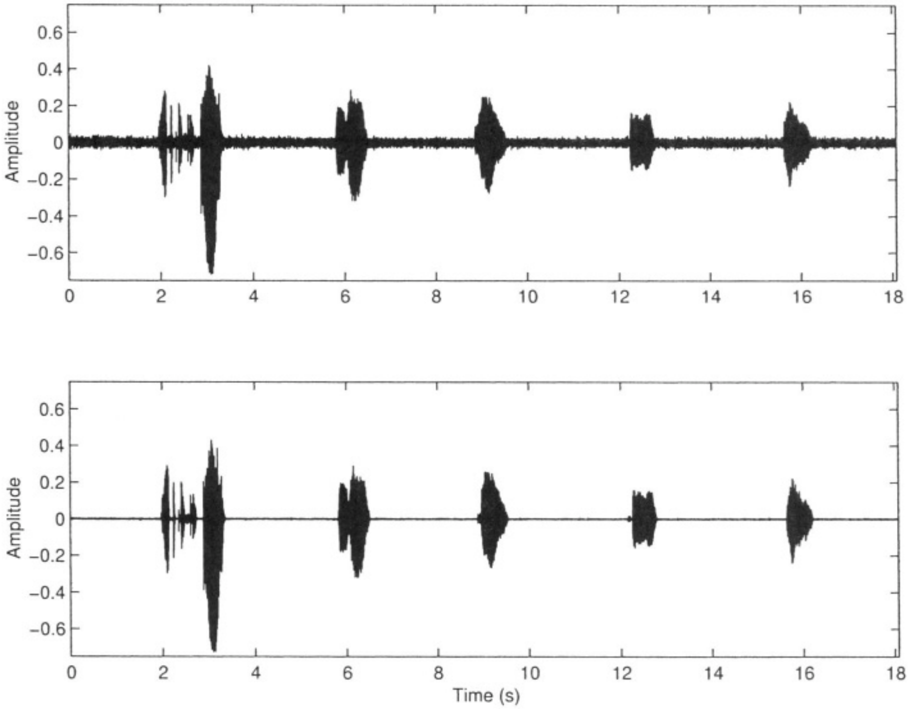


Figure 4.4 Speech time series for the noise reduction example. Original (top) and noise-reduced (bottom) time series.

Notes

1. The estimated instantaneous *a priori* SNR is the ratio $|\hat{S}(k,m)|^2/|V(k,m)|^2$.

References

[1] M. R. Schroeder, U.S. Patent No. 3,180,936, filed Dec. 1, 1960, issued Apr. 27, 1965.

[2] M. R. Schroeder, U.S. Patent No. 3,403,224, filed May 28, 1965, issued Sep. 24, 1968.

[3] M. M. Sondhi and S. Sievers, AT&T Bell Laboratories Internal Report (unpublished), Dec. 1964.

[4] M. M. Sondhi, C. E. Schmidt, and L. R. Rabiner, "Improvingthe quality of a noisy speech signal," *Bell Syst. Techn. J.*, vol. 60, Oct. 1981.

[5] R. E. Crochiere and L. R. Rabiner, *Multimte Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1983.

[6] N. Wiener, *Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications.* New York: Wiley, 1949.

[7] H. L. Van Trees, *Detection, Estimation, and Modulation Theory, Part I.* New York: John Wiley & Sons, 1968.

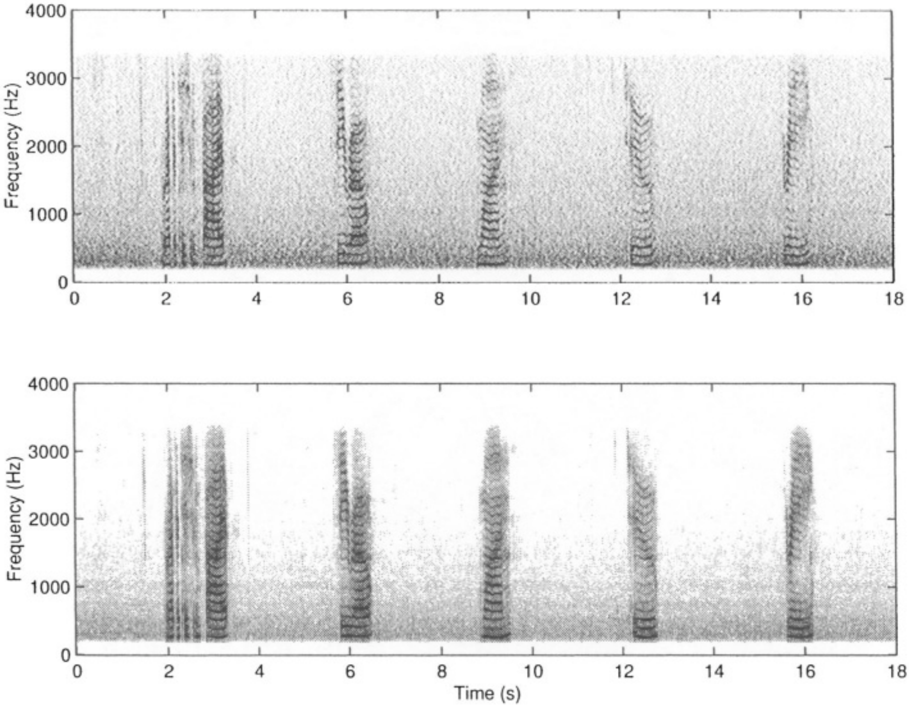


Figure 4.5 **Spectrograms corresponding to speech time series in Fig. 4.4. Original (top) and noise-reduced (bottom) time series. Spectrograms displayed using a dynamic range of 80 dB.**

[8] M. R. Portnoff, "Time-frequency representation of digital signals and systems based on short-time Fourier analysis," *IEEE Trans. Acoust. Speech Signal Process.,* vol. ASSP-28, pp. 55-69, Feb. 1980.

[9] J. B. Allen, "Short-time spectral analysis, synthesis and modification by discrete Fourier transform," *IEEE Trans. Acoust. Speech Signal Process.,* vol. ASSP-25, pp. 235-238, June 1977.

[10] L. R. Rabiner, M. R. Sambur, and C. E. Schmidt, "Applications of a nonlinear smoothing algorithm to speech processing," *IEEE Trans. Acoust. Speech Signal Process.,* vol. ASSP-23, Dec. 1975.

[11] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust.. Speech, Signal Proc.,* vol. ASSP-27, Apr. 1979.

[12] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.,* vol. ASSP-32, Dec. 1984.

[13] R. J. McAulay and M. L. Malpass, "Speech enhancement using a soft-decision noise suppression filter," *IEEE Trans. Acoust., Speech, Signal Process.,* vol. ASSP-28, Apr. 1980.

[14] J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. of the IEEE,* vol. 67, Dec. 1979.

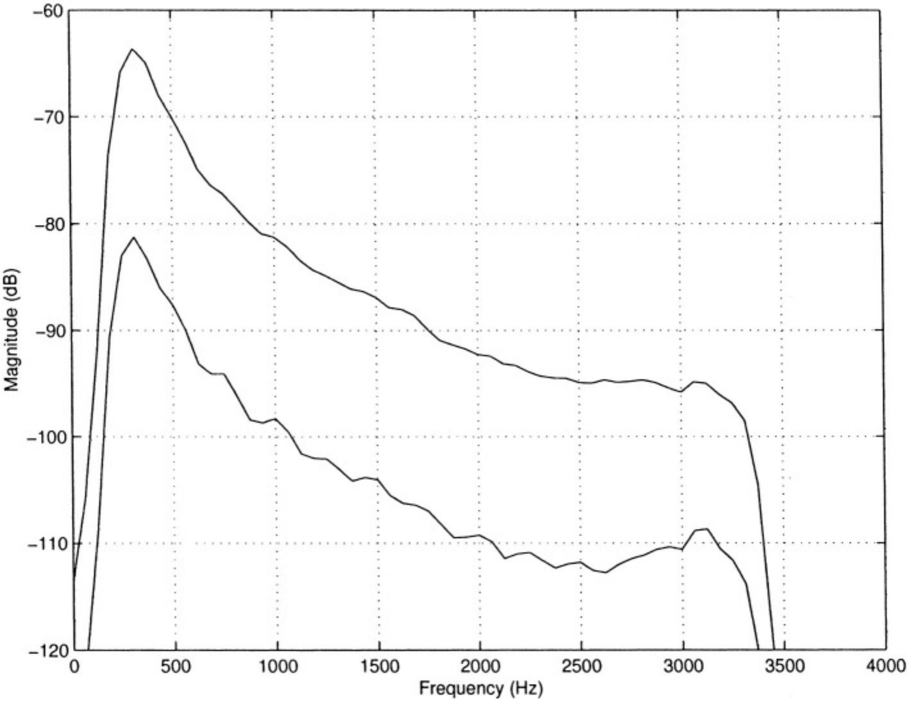


Figure 4.6 Noisy (top) and noise-reduced (bottom) power spectral density corresponding to the time series in Fig. 4.4. Power spectral densities computed from the time series in Fig. 4.4 over the interval $10s - 12s$.

[15] M. R. Weiss, E. Aschkenasy, and T. W. Parsons, "Processing speech signals to attenuate interference," in *Proc. IEEE Symposium on Speech Recognition,* Carnegie-Mellon Univ., Apr. 15-19, 1974, pp. 292-295.

[16] O. Cappé, "Elimination of the musical noise phenomenon with the Ephraim and Malah noise suppressor," *IEEE Trans. Speech Audio Process.,* vol. 2, Apr. 1994.

[17] W. Etter and G. S. Moschytz, "Noise reduction by noise-adaptive spectral magnitude expansion," *J. Audio Eng. Soc.,* vol. 42, May 1994.

[18] B. M. Helf and P. L. Chu, "Reduction of background noise for speech enhancement," U.S. Patent No. 5,550,924, Mar. 13, 1995.

[19] E. J. Diethorn, "A subband noise-reduction method for enhancing speech in telephony & teleconferencing," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics,* Mohonk Mountain House, New Paltz, NY, Oct. 19-22, 1997.

[20] P. Vary, "Noise suppression by spectral magnitude estimation – mechanism and theoretical limits,"*Signal Processing,* vol. 8, pp. 387-400, July 1985.

[21] Y. Ephraim, D. Malah, and B. H. Juang, "On the application of hidden Markov models for enhancing noisy speech," *IEEE Trans. Acoust., Speech, Signal Process.,* vol. ASSP-37, pp. 1846-1856, Dec. 1989.

[22] H. Sameti, H. Sheikhzadeh, and Li Deng, “HMM-based strategies for enhancement of speech signals embedded in nonstationary noise,” *IEEE Trans. Speech Audio Process.,* vol. 6, Sep. 1998.

[23] B. L. Sim, Y. C. Tong, and J. S. Chang, “A parametric formulation of the generalized spectral subtraction method,” *IEEE Trans. Speech Audio Process.,* vol. 6, July 1998.

[24] D. E. Tsoukalas and J. N. Mourjopoulos, “Speech enhancement based on audible noise suppression,” *IEEE Trans. Speech Audio Process.,* vol. 5, Nov. 1997.

[25] F. Plante, G. Meyer, and W. A. Ainsworth, “Improvement of speech spectrogram accuracy by the method of reassignment,” *IEEE Trans. Speech Audio Process.,* vol. 6, May 1998.

[26] R. H. Erving, W. A. Ford, and R. Miller, U.S. Patent No. 5,007,046, filed Dec. 28, 1988, issued Apr. 9, 1991.

This page intentionally left blank

II

ACOUSTIC ECHO CANCELLATION

This page intentionally left blank

Chapter 5

ADAPTIVE ALGORITHMS FOR MIMO ACOUSTIC ECHO CANCELLATION

Jacob Benesty
Université du Québec, INRS-EMT
benesty@inrs-emt.uquebec.ca

Tomas Gänsler
Agere Systems
gaensler@agere.com

Yiteng (Arden) Huang
Bell Laboratories, Lucent Technologies
arden@research.bell-labs.com

Markus Rupp
TU Wien, Institute for Communication and RF Engineering
mrupp@nt.tuwien.ac.at

Abstract The first thing that comes in mind when we talk about acoustic echo cancellation is adaptive filtering. In this chapter, we discuss a large number of multichannel adaptive algorithms, both in time and frequency domains. This discussion will be developed in the context of multichannel acoustic echo cancellation where we have to identify a multiple-input multiple-output (MIMO) system (e.g., room acoustic impulse responses).

Keywords: Acoustic Echo Cancellation, Multichannel, Adaptive Algorithms, LMS, APA, RLS, FRLS, Exponentiated, MIMO, Frequency-Domain

1. INTRODUCTION

All today's teleconferencing systems are hands-free and single-channel (meaning that there is only one microphone and one loudspeaker). In the near future, we expect that multichannel systems (with at least two loudspeakers and at least one microphone) will be available to customers, therefore providing a realistic presence that single-channel systems cannot offer.

In hands-free systems, the coupling between loudspeakers and microphones can be very strong and this can generate important echoes that eventually make the system completely unstable (e.g., the system starts howling). Therefore, multichannel acoustic echo cancelers (MCAECs) are absolutely necessary for full-duplex communication [1]. Let P and Q be respectively the numbers of loudspeakers and microphones. For a teleconferencing system, the MCAECs consist of PQ adaptive filters aiming at identifying PQ echo paths from P loudspeakers to Q microphones. This scheme is, in fact, a multiple-input multiple-output (MIMO) system. We assume that the teleconferencing system is organized between two rooms: the "transmission" and "receiving" rooms. The transmission room is sometimes referred to as the far-end and the receiving room as the near-end. So each room needs an MCAEC for each microphone. Thus, multichannel acoustic echo cancellation consists of a direct identification of an unknown linear MIMO system.

Although conceptually very similar, multichannel acoustic echo cancellation (MCAEC) is fundamentally different from traditional mono echo cancellation in one respect: a straightforward generalization of the mono echo canceler would not only have to track changing echo paths in the receiving room, *but also in the transmission room*! For example, the canceler would have to reconverge if one talker stops talking and another starts talking at a different location in the transmission room. There is no adaptive algorithm that can track such a change sufficiently fast and this scheme therefore results in poor echo suppression. Thus, a generalization of the mono AEC in the multichannel case does not result in satisfactory performance.

The theory explaining the problem of MCAEC was described in [1] and [2]. The fundamental problem is that the multiple channels may carry linearly related signals which in turn may make the normal equations to be solved by the adaptive algorithm singular. This implies that there is no unique solution to the equations but an infinite number of solutions, and it can be shown that all but the true one depend on the impulse responses of the transmission room. As a result, intensive studies have been made of how to handle this properly. It was shown in [2] that the only solution to the nonuniqueness problem is to reduce the coherence between the different loudspeaker signals, and an efficient low complexity method for this purpose was also given.

Lately, attention has been focused on the investigation of other methods that decrease the cross-correlation between the channels in order to get well behaved estimates of the echo paths [3], [4], [5], [6], [7], [8]. The main problem is how to reduce the coherence sufficiently without affecting the stereo perception and the sound quality.

The performance of the MCAEC is more severely affected by the choice of the adaptive algorithm than the monophonic counterpart [9], [10]. This is easily recognized since the performance of most adaptive algorithms depends on the condition number of the input signal covariance matrix. In the multichannel case, the condition number is very high; as a result, algorithms such as the least-mean-square (LMS) or the normalized LMS (NLMS), which do not take into account the cross-correlation among all the input signals, converge very slowly to the true solution. It is therefore highly interesting to study multichannel adaptive filtering algorithms.

In this chapter, we develop a general framework for multichannel adaptive filters with the purpose to improve their performance in time and frequency domains. We also investigate a recently proposed class of adaptive algorithms that exploit sparsity of room acoustic impulse responses. These algorithms are very interesting both from theoretical and practical standpoints since they converge and track much better than the NLMS algorithm for example.

2. NORMAL EQUATIONS AND IDENTIFICATION OF A MIMO SYSTEM

We first derive the normal equations of a multiple-input multiple-output (MIMO) system.

2.1 NORMAL EQUATIONS

We assume that we have a MIMO system with P inputs (loudspeakers) and Q outputs (microphones). We also assume that the MIMO system (a room in our context) is linear and time-invariant. Acoustic echo cancellation consists of identifying P echo paths at each microphone so that in total, PQ echo paths need to be estimated. We have Q output (microphone) signals (see Fig. 5.1):

$$\begin{aligned} y_q(n) &= \sum_{p=1}^{P} \mathbf{h}_{pq}^T \mathbf{x}_p(n) + b_q(n), \\ q &= 1, 2, ..., Q, \end{aligned} \tag{5.1}$$

where superscript ^{T} denotes transpose of a vector or a matrix,

$$\mathbf{h}_{pq} = \begin{bmatrix} h_{pq,0} & h_{pq,1} & \cdots & h_{pq,L-1} \end{bmatrix}^T$$

is the echo path – of length L – between loudspeaker p and microphone q,

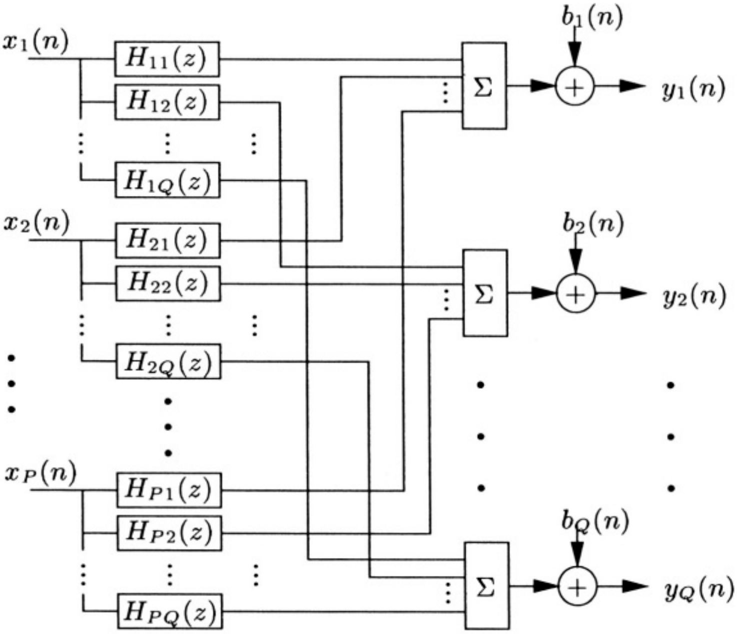


Figure 5.1 A MIMO system consisting of P inputs and Q outputs.

$$\mathbf{x}_p(n) = \begin{bmatrix} x_p(n) & x_p(n-1) & \cdots & x_p(n-L+1) \end{bmatrix}^T,$$
$$p = 1, 2, ..., P,$$

is the pth reference (loudspeaker) signal (also called the far-end speech), and $b_q(n)$ is the near-end noise added at microphone q, assumed to be uncorrelated with the far-end speech. We define the error signal at time n for microphone q as

$$\begin{aligned} e_q(n) &= y_q(n) - \hat{y}_q(n) \\ &= y_q(n) - \sum_{p=1}^{P} \hat{\mathbf{h}}_{pq}^T \mathbf{x}_p(n), \end{aligned} \tag{5.2}$$

where

$$\hat{\mathbf{h}}_{pq} = \begin{bmatrix} \hat{h}_{pq,0} & \hat{h}_{pq,1} & \cdots & \hat{h}_{pq,L-1} \end{bmatrix}^T$$

are the model filters. It is more convenient to define an error signal vector for all the microphones:

$$\begin{aligned} \mathbf{e}(n) &= \mathbf{y}(n) - \hat{\mathbf{y}}(n) \\ &= \mathbf{y}(n) - \hat{\mathbf{H}}^T \mathbf{x}(n), \end{aligned} \tag{5.3}$$

where

$$\begin{aligned}
\mathbf{y}(n) &= \mathbf{H}^T\mathbf{x}(n) + \mathbf{b}(n), \\
\mathbf{H} &= \begin{bmatrix} \mathbf{h}_{11} & \mathbf{h}_{12} & \cdots & \mathbf{h}_{1Q} \\ \mathbf{h}_{21} & \mathbf{h}_{22} & \cdots & \mathbf{h}_{2Q} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}_{P1} & \mathbf{h}_{P2} & \cdots & \mathbf{h}_{PQ} \end{bmatrix}, \\
\mathbf{b}(n) &= \begin{bmatrix} b_1(n) & b_2(n) & \cdots & b_Q(n) \end{bmatrix}^T, \\
\mathbf{e}(n) &= \begin{bmatrix} e_1(n) & e_2(n) & \cdots & e_Q(n) \end{bmatrix}^T, \\
\mathbf{y}(n) &= \begin{bmatrix} y_1(n) & y_2(n) & \cdots & y_Q(n) \end{bmatrix}^T, \\
\hat{\mathbf{y}}(n) &= \begin{bmatrix} \hat{y}_1(n) & \hat{y}_2(n) & \cdots & \hat{y}_Q(n) \end{bmatrix}^T, \\
\hat{\mathbf{H}} &= \begin{bmatrix} \hat{\mathbf{h}}_{11} & \hat{\mathbf{h}}_{12} & \cdots & \hat{\mathbf{h}}_{1Q} \\ \hat{\mathbf{h}}_{21} & \hat{\mathbf{h}}_{22} & \cdots & \hat{\mathbf{h}}_{2Q} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\mathbf{h}}_{P1} & \hat{\mathbf{h}}_{P2} & \cdots & \hat{\mathbf{h}}_{PQ} \end{bmatrix}, \\
\mathbf{x}(n) &= \begin{bmatrix} \mathbf{x}_1^T(n) & \mathbf{x}_2^T(n) & \cdots & \mathbf{x}_P^T(n) \end{bmatrix}^T.
\end{aligned}$$

Having written the error signal, we now define the recursive least-squares error criterion with respect to the modelling filters:

$$\begin{aligned}
J(n) &= \sum_{i=0}^{n} \lambda^{n-i}\mathbf{e}^T(i)\mathbf{e}(i) \qquad (5.4) \\
&= \sum_{q=1}^{Q}\sum_{i=0}^{n} \lambda^{n-i}e_q^2(i) \\
&= \sum_{q=1}^{Q} J_q(n),
\end{aligned}$$

where λ $(0 < \lambda < 1)$ is a forgetting factor. The minimization of (5.4) leads to the multichannel normal equations:

$$\mathbf{R}_{xx}(n)\hat{\mathbf{H}}(n) = \mathbf{R}_{xy}(n), \qquad (5.5)$$

where

$$\begin{aligned}\mathbf{R}_{xx}(n) &= \sum_{i=0}^{n} \lambda^{n-i}\mathbf{x}(i)\mathbf{x}^T(i) \\ &= \begin{bmatrix} \mathbf{R}_{11}(n) & \mathbf{R}_{12}(n) & \cdots & \mathbf{R}_{1P}(n) \\ \mathbf{R}_{21}(n) & \mathbf{R}_{22}(n) & \cdots & \mathbf{R}_{2P}(n) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P1}(n) & \mathbf{R}_{P2}(n) & \cdots & \mathbf{R}_{PP}(n) \end{bmatrix} \end{aligned} \qquad (5.6)$$

is an estimate of the input signal covariance matrix – of size $(PL \times PL)$, and

$$\mathbf{R}_{xy}(n) = \sum_{i=0}^{n} \lambda^{n-i}\mathbf{x}(i)\mathbf{y}^T(i) \qquad (5.7)$$

is an estimate of the cross-correlation matrix – of size $(PL \times Q)$ – between $\mathbf{x}(n)$ and $\mathbf{y}^T(n)$.

It can easily be seen that the multichannel normal equations (5.5) can be decomposed in Q independent normal equations, each one corresponding to a microphone signal:

$$\mathbf{R}_{xx}(n)\hat{\mathbf{h}}_q(n) = \mathbf{r}_{xy,q}(n),\ q = 1, 2, ..., Q, \qquad (5.8)$$

where $\hat{\mathbf{h}}_q(n)$ [resp. $\mathbf{r}_{xy,q}(n)$] is the qth column ofmatrix $\hat{\mathbf{H}}(n)$ [resp. $\mathbf{R}_{xy}$(n)]. This result implies that minimizing $J(n)$ or minimizing each $J_q(n)$ independently gives the same results. This makes sense from an identification point of view, since the identification of the impulse responses for one microphone is completely independent of the others.

2.2 THE NONUNIQUENESS PROBLEM

In many situations, the signals $x_p(n)$ are generated from a unique source $s(n)$, so that:

$$x_p(n) = \mathbf{g}_p^T\mathbf{s}(n),\ p = 1, 2, ..., P, \qquad (5.9)$$

where

$$\mathbf{g}_p = \begin{bmatrix} g_{p,0} & g_{p,1} & \cdots & g_{p,L-1} \end{bmatrix}^T$$

is the impulse response between the source andmicrophone p in the transmission room in the case of a teleconferencing system [2]. Therefore the signals $x_p(n)$ are linearly related and we have the following $[P(P - 1)/2]$ relations [2]:

$$\begin{gathered}\mathbf{x}_p^T(n)\mathbf{g}_i = \mathbf{x}_i^T(n)\mathbf{g}_p, \\ i, p = 1, 2, ..., P,\ i \neq p.\end{gathered} \qquad (5.10)$$

Indeed, since $x_p = s * g_p$, therefore $x_p * g_i = s * g_p * g_i = x_i * g_p$ (the symbol $*$ is the linear convolution operator). Now, consider the following vector:

$$\mathbf{u} = \begin{bmatrix} \sum_{p=2}^{P} \zeta_p \mathbf{g}_p^T & -\zeta_2 \mathbf{g}_1^T & \cdots & -\zeta_P \mathbf{g}_1^T \end{bmatrix}^T,$$

where ζ_p are arbitrary factors. We can verify using (5.10) that $\mathbf{R}_{xx}(n)\mathbf{u} = \mathbf{0}_{PL\times 1}$, so $\mathbf{R}_{xx}(n)$ is not invertible. Vector $\mathbf{u}$ represents the nullspace of matrix $\mathbf{R}_{xx}(n)$. The dimension of this nullspace depends of the number of inputs and is equal to $(P - 2)L + 1$ (for $P \geq 2$). So the problem becomes worse as P increases. Thus, there is no unique solution to the problem and an adaptive algorithm will drift to any one of many possible solutions, which can be very different from the "true" desired solution $\hat{\mathbf{h}}_{pq} = \mathbf{h}_{pq}$. These nonunique "solutions" are dependent on the impulse responses in the transmission room:

$$\hat{\mathbf{h}}_{1q} = \mathbf{h}_{1q} + \beta \sum_{p=2}^{P} \zeta_p \mathbf{g}_p, \tag{5.11}$$

$$\hat{\mathbf{h}}_{pq} = \mathbf{h}_{pq} - \beta \zeta_p \mathbf{g}_1,\ p = 2, ..., P, \tag{5.12}$$

where β is an arbitrary factor. This, of course, is intolerable because $\mathbf{g}_p$ can change instantaneously, for example, as one person stops talking and another starts [1], [2].

2.3 THE IMPULSE RESPONSE TAIL EFFECT

We first define an important measure that is very useful for MCAEC.

Definition: The quantity

$$\frac{\|\mathbf{h}_q - \hat{\mathbf{h}}_q\|}{\|\mathbf{h}_q\|},\ q = 1, 2, ..., Q, \tag{5.13}$$

where $\|\cdot\|$ denotes the two-norm vector, is called the *normalized misalignment* and measures the mismatch between the impulse responses of the receiving room and the modelling filters. In the multichannel case, it is possible to have good echo cancellation even when the misalignment is large. However, in such a case, the cancellation will degrade if the $\mathbf{g}_p$ change. A main objective of MCAEC research is to avoid this problem.

Actually, for the practical case when the length of the adaptive filters is smaller than the length of the impulse responses in the transmission room, there is a unique solution to the normal equation, although the covariance matrix is very ill-conditioned.

On the other hand, we can easily show by using the classical normal equations that if the length of the adaptive filters is smaller than the length of the impulse responses in the receiving room, we introduce an important bias in the coefficients of these filters because of the strong cross-correlation between the input signals and the large condition number of the covariance matrix [2]. So in practice, we may have poor misalignment even if there is a unique solution to the normal equations.

The only way to decrease the misalignment is to partially decorrelate two-by-two the *P* input (loudspeaker) signals. Next, we summarize a number of approaches that have been developed recently for reducing the cross-correlation.

2.4 SOME DIFFERENT SOLUTIONS FOR DECORRELATION

If we have *P* different channels, we need to decorrelate them partially and mutually. In the following, we show how to partially decorrelate two channels. The same process should be applied for all the channels. It is well-known that the coherence magnitude between two processes is equal to 1 if and only if they are linearly related. In order to weaken this relation, some non-linear or time-varying transformation of the stereo channels has to be made. Such a transformation reduces the coherence and hence the condition number of the covariance matrix, thereby improving the misalignment. However, the transformation has to be performed cautiously so that it is inaudible and has no effect on stereo perception.

A simple nonlinear method that gives good performance uses a half-wave rectifier [2], so that the nonlinearly transformed signal becomes

$$x_p'(n) = x_p(n) + \alpha \frac{x_p(n) + |x_p(n)|}{2}, \tag{5.14}$$

where α is a parameter used to control the amount of nonlinearity. For this method, there can only be a linear relation between the nonlinearly transformed channels if $\forall n, x_1(n) \geq 0$ and $x_2(n) \geq 0$ or if we have $ax_1(n-\tau_1) = x_2(n-\tau_2)$ with $a > 0$. In practice however, these cases never occur because we always have zero-mean signals and $\mathbf{g}_1$, $\mathbf{g}_2$ are in practice never related by just a simple delay.

An improved version of this technique is to use positive and negative half-wave rectifiers on each channel respectively,

$$x_1'(n) = x_1(n) + \alpha \frac{x_1(n) + |x_1(n)|}{2}, \tag{5.15}$$

$$x_2'(n) = x_2(n) + \alpha \frac{x_2(n) - |x_2(n)|}{2}. \tag{5.16}$$

This principle removes the linear relation even in the special signal cases given above.

Experiments show that stereo perception is not affected by the above methods even with α as large as 0.5. Also, the distortion introduced for speech is hardly audible because of the nature of the speech signal and psychoacoustic masking effects [11]. This is explained by the following three reasons. First, the distorted signal $x'_p(n)$ depends only on the instantaneous value of the original signal $x_p(n)$ so that during periods of silence, no distortion is added. Second, the periodicity remains unchanged. Third, for voiced sounds, the harmonic structure of the signal induces “self-masking” of the harmonic distortion components. This kind of distortion is also acceptable for some music signals but may be objectionable for pure tones.

Other types of nonlinearities for decorrelating speech signals have also been investigated and compared [12]. The results indicate that, of the several nonlinearities considered, ideal half-wave rectification and smoothed half-wave rectification appear to be the best choices for speech. For music, the nonlinearity parameter of the ideal rectifier must be readjusted. The smoothed rectifier does not require this readjustment but is a little more complicated to implement.

In [6] a similar approach with non-linearities is proposed. The idea is expanded so that four adaptive filters operate on different non-linearly processed signals to estimate the echo paths. These non-linearities are chosen such that the input signals of two of the adaptive filters are independent, which thus represent a “perfect” decorrelation. Tap estimates are then copied to a fixed two-channel filter which performs the echo cancellation with the unprocessed signals. The advantage of this method is that the NLMS algorithm could be used instead of more sophisticated algorithms.

Another approach that makes it possible to use the NLMS algorithm is to decorrelate the channels by means of complementary comb filtering [1], [13]. The technique is based on removing the energy in a certain frequency band of the speech signal in one channel. This means the coherence would become zero in this band and thereby results in fast alignment of the estimate even when using the NLMS algorithm. Energy is removed complementarily between the channels so that the stereo perception is not severely affected for frequencies above 1 kHz. However, this method must be combined with some other decorrelation technique for lower frequencies [14].

Two methods based on introducing time-varying filters in the transmission path were presented in [7], [8]. In [7], left and right signals are filtered through two independent time-varying first-order all-pass filters. Stochastic time-variation is introduced by making the pole position of the filter a random walk process. The actual position is limited by the constraints of stability and inaudibility of the introduced distortion. While significant reduction in correlation can be achieved for higher frequencies with the imposed constraints, the lower frequencies are still fairly unaffected by the time-variation. In [8], a periodically varying filter is applied to one channel so that the signal is either delayed by one sample or passed through without delay. A transition zone between the delayed and non-delayed state is also employed in order to reduce audible discontinuities. This method may also affect the stereo perception.

Although the idea of adding independent perceptually shaped noise to the channels was mentioned in [1], [2], thorough investigations of the actual benefit of the technique was not presented. Results regarding variants of this idea can be found in [4], [5]. A pre-processing unit estimating the masking threshold and adding an appropriate amount of noise was proposed in [5]. It was also noted that adding a masked noise to each channel may affect the spatialization of the sound even if the noise is inaudible at each channel separately. This effect can be controlled through correction of the masking threshold when appropriate. In [4], the improvement of misalignment was studied in the SAEC when a perceptual audio coder was added in the transmission path. Reduced correlation between the channels was shown by means of coherence analysis, and improved convergence rate of the adaptive algorithm was observed. A low-complexity method for achieving additional decorrelation by modifying the decoder was also proposed. The encoder cannot quantize every single frequency band optimally due to rate constraints. This has the effect that there is a margin on the masking threshold which can be exploited. In the presented method, the masking threshold is estimated from the modified discrete cosine transform (MDCT) coefficients delivered by the encoder, and an appropriate inaudible amount of decorrelating noise is added to the signals.

In the rest of this chapter, we suppose that one of the previous decorrelation methods is used so the normal equations have a unique solution. However, the input signals can still be highly correlated, therefore requiring special treatment.

3. THE CLASSICAL AND FACTORIZED MULTICHANNEL RLS

From the normal equations (5.8), we easily derived the classical update equations for the multichannel recursive least-squares (RLS):

$$e_q(n) = y_q(n) - \hat{\mathbf{h}}_q^T(n-1)\mathbf{x}(n), \tag{5.17}$$

$$\hat{\mathbf{h}}_q(n) = \hat{\mathbf{h}}_q(n-1) + \mathbf{R}_{xx}^{-1}(n)\mathbf{x}(n)e_q(n). \tag{5.18}$$

Note that the Kalman gain $\mathbf{k}(n) = \mathbf{R}_{xx}^{-1}(n)\mathbf{x}(n)$ is the same for all the microphone signals $q = 1, 2, \ldots, Q$. This is important, even though we have Q update equations, the Kalman vector needs to be computed only one time per iteration. Using the matrix inversion lemma, we obtain the following recursive equation for the inverse of the covariance matrix:

$$\mathbf{R}_{xx}^{-1}(n) = \lambda^{-1}\mathbf{R}_{xx}^{-1}(n-1) - \frac{\lambda^{-2}\mathbf{R}_{xx}^{-1}(n-1)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{R}_{xx}^{-1}(n-1)}{1+\lambda^{-1}\mathbf{x}^T(n)\mathbf{R}_{xx}^{-1}(n-1)\mathbf{x}(n)}. \tag{5.19}$$

Another way to write the multichannel RLS is to first factorize the covariance matrix inverse $\mathbf{R}_{xx}^{-1}(n)$.

Consider the following variables:

$$\begin{aligned}
\mathbf{z}_p(n) &= \sum_{j=1}^{P} \mathbf{C}_{pj}\mathbf{x}_j(n) \\
&= \mathbf{x}_p(n) + \sum_{j=1, j\neq p}^{P} \mathbf{C}_{pj}\mathbf{x}_j(n) \\
&= \mathbf{x}_p(n) - \hat{\mathbf{x}}_p(n), \; p = 1, ..., P, \qquad (5.20)
\end{aligned}$$

with $\mathbf{C}_{pp} = \mathbf{I}_{L\times L}$ and $\hat{\mathbf{x}}_p(n) = -\sum_{j=1,j\neq p}^{P} \mathbf{C}_{pj}\mathbf{x}_j(n)$. Matrices $\mathbf{C}_{pj}$ are the cross-interpolators obtained by minimizing

$$J_{z_p}(n) = \sum_{i=0}^{n} \lambda^{n-i}\mathbf{z}_p^T(i)\mathbf{z}_p(i), \; p = 1, ..., P, \qquad (5.21)$$

and $\mathbf{z}_p(n)$ are the cross-interpolation error vectors.

A general factorization of $\mathbf{R}_{xx}^{-1}(n)$ can be stated as follows:

Lemma 1:

$$\mathbf{R}_{xx}^{-1}(n) = \begin{bmatrix} \mathbf{R}_1^{-1}(n) & \mathbf{0}_{L\times L} & \cdots & \mathbf{0}_{L\times L} \\ \mathbf{0}_{L\times L} & \mathbf{R}_2^{-1}(n) & \cdots & \mathbf{0}_{L\times L} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{L\times L} & \mathbf{0}_{L\times L} & \cdots & \mathbf{R}_P^{-1}(n) \end{bmatrix} \times \begin{bmatrix} \mathbf{I}_{L\times L} & \mathbf{C}_{12}(n) & \cdots & \mathbf{C}_{1P}(n) \\ \mathbf{C}_{21}(n) & \mathbf{I}_{L\times L} & \cdots & \mathbf{C}_{2P}(n) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{P1}(n) & \mathbf{C}_{P2}(n) & \cdots & \mathbf{I}_{L\times L} \end{bmatrix}, \qquad (5.22)$$

where

$$\mathbf{R}_p(n) = \sum_{j=1}^{P} \mathbf{C}_{pj}(n)\mathbf{R}_{jp}(n), \; p = 1, 2, ..., P. \qquad (5.23)$$

Proof: The proof is rather straightforward by multiply ing both sides of (5.22) by $\mathbf{R}_{xx}(n)$ and showing that the result of the right-hand side is equal to the identity matrix with the help of (5.21).

Example: P = 2: In this case, we have:

$$\mathbf{z}_1(n) = \mathbf{x}_1(n) + \mathbf{C}_{12}\mathbf{x}_2(n), \qquad (5.24)$$

$$\mathbf{z}_2(n) = \mathbf{x}_2(n) + \mathbf{C}_{21}\mathbf{x}_1(n), \qquad (5.25)$$

where

$$\begin{aligned}\mathbf{C}_{12}(n) &= -\mathbf{R}_{12}(n)\mathbf{R}_{22}^{-1}(n), &(5.26)\\ \mathbf{C}_{21}(n) &= -\mathbf{R}_{21}(n)\mathbf{R}_{11}^{-1}(n), &(5.27)\end{aligned}$$

are the cross-interpolators obtained by minimizing $\sum_{i=0}^{n} \lambda^{n-i}\mathbf{z}_1^T(i)\mathbf{z}_1(i)$ and $\sum_{i=0}^{n} \lambda^{n-i}\mathbf{z}_2^T(i)\mathbf{z}_2(i)$. Hence:

$$\begin{aligned}\mathbf{R}^{-1}(n) &= \begin{bmatrix} \mathbf{R}_1^{-1}(n) & \mathbf{0}_{L\times L} \\ \mathbf{0}_{L\times L} & \mathbf{R}_2^{-1}(n) \end{bmatrix} \times \\ &\begin{bmatrix} \mathbf{I}_{L\times L} & -\mathbf{R}_{12}(n)\mathbf{R}_{22}^{-1}(n) \\ -\mathbf{R}_{21}(n)\mathbf{R}_{11}^{-1}(n) & \mathbf{I}_{L\times L} \end{bmatrix}, \qquad (5.28)\end{aligned}$$

where

$$\begin{aligned}\mathbf{R}_1(n) &= \mathbf{R}_{11}(n) - \mathbf{R}_{12}(n)\mathbf{R}_{22}^{-1}(n)\mathbf{R}_{21}(n), &(5.29)\\ \mathbf{R}_2(n) &= \mathbf{R}_{22}(n) - \mathbf{R}_{21}(n)\mathbf{R}_{11}^{-1}(n)\mathbf{R}_{12}(n), &(5.30)\end{aligned}$$

are the cross-interpolation error energy matrices or the Schur complements of $\mathbf{R}_{xx}(n)$ with respect to $\mathbf{R}_{22}(n)$ and $\mathbf{R}_{11}(n)$.

From the above result (Lemma 1), we deduce the factorized multichannel RLS:

$$\begin{aligned}&\hat{\mathbf{h}}_{pq}(n) = \hat{\mathbf{h}}_{pq}(n-1) + \mathbf{R}_p^{-1}(n)\mathbf{z}_p(n)e_q(n), \qquad (5.31)\\ &p = 1,2,...,P, \; q = 1,2,...,Q.\end{aligned}$$

4. THE MULTICHANNEL FAST RLS

Because RLS has so far proven to perform better than other algorithms in the MCAEC application [15] a fast calculation scheme of a multichannel version is presented in this section. Compared to standard RLS it has a much lower complexity, $6P^2L+2PL$ multiplications [instead of $O(P^2L^2)$] for one system output. This algorithm is a numerically stabilized version of the algorithm proposed in [16]. Some extra stability control has to be added so that the algorithm behaves well for a non-stationary speech signal. The following has to be defined:

$$\begin{aligned}\chi(n) &= [x_1(n)\; x_2(n)\; \cdots\; x_P(n)]^T, \; (P\times 1), &(5.32)\\ \tilde{\mathbf{x}}(n) &= [\chi^T(n)\; \chi^T(n-1)\; \cdots\; \chi^T(n-L+1)]^T, &(5.33)\\ &\quad (PL\times 1),\\ \tilde{\mathbf{h}}_q(n) &= [\hat{h}_{1q,0}(n)\; \hat{h}_{2q,0}(n)\; \cdots\; \hat{h}_{(P-1)q,L-1}(n)\; \hat{h}_{Pq,L-1}(n)]^T,\\ &\quad (PL\times 1). &(5.34)\end{aligned}$$

Note that the channels of the filter and state-vector $[\tilde{\mathbf{x}}(n)]$ are interleaved in this algorithm. Defined also:

- $\mathbf{A}(n)$, $\mathbf{B}(n)$ = Forward and backward prediction filter matrices, $(PL \times P)$,
- $\mathbf{E}_{\mathrm{A}}(n)$, $\mathbf{E}_{\mathrm{B}}(n)$ = Forward and backward prediction error energy matrices, $(P \times P)$,
- $\mathbf{e}_{\mathrm{A}}(n)$, $\mathbf{e}_{\mathrm{B}}(n)$ = Forward and backward prediction error vectors, $(P\times 1)$,
- $\mathbf{k}'(n) = \mathbf{R}_{xx}^{-1}(n-1)\tilde{\mathbf{x}}(n)$ = *a priori* Kalman vector, $(PL \times 1)$,
- $\varphi(n)$ = Maximum likelihood related variable, (1×1),
- $\kappa \in [1.5, 2.5]$, Stabilization parameter, (1×1),
- $\lambda \in (0, 1]$, Forgetting factor, (1×1).

The multichannel fast RLS (FRLS) is then:

$$\begin{aligned}
& \quad \textit{Prediction}: \\
\mathbf{e}_{\mathrm{A}}(n) &= \chi(n) - \mathbf{A}^T(n-1)\tilde{\mathbf{x}}(n-1), \quad (P \times 1), \\
\varphi_1(n) &= \varphi(n-1) + \mathbf{e}_{\mathrm{A}}^T(n)\mathbf{E}_{\mathrm{A}}^{-1}(n-1)\mathbf{e}_{\mathrm{A}}(n), \quad (1 \times 1), \\
\begin{bmatrix} \mathbf{t}(n) \\ \mathbf{m}(n) \end{bmatrix} &= \begin{bmatrix} \mathbf{0}_{P\times 1} \\ \mathbf{k}'(n-1) \end{bmatrix} + \begin{bmatrix} \mathbf{I}_{P\times P} \\ -\mathbf{A}(n-1) \end{bmatrix} \mathbf{E}_{\mathrm{A}}^{-1}(n-1)\mathbf{e}_{\mathrm{A}}(n), \\
& \quad ((PL+P) \times P), \\
\mathbf{E}_{\mathrm{A}}(n) &= \lambda[\mathbf{E}_{\mathrm{A}}(n-1) + \mathbf{e}_{\mathrm{A}}(n)\mathbf{e}_{\mathrm{A}}^T(n)/\varphi(n-1)], \quad (P \times P), \\
\mathbf{A}(n) &= \mathbf{A}(n-1) + \mathbf{k}'(n-1)\mathbf{e}_{\mathrm{A}}^T(n)/\varphi(n-1), \quad (PL \times P), \\
\mathbf{e}_{\mathrm{B}_1}(n) &= \mathbf{E}_{\mathrm{B}}(n-1)\mathbf{m}(n), \quad (P \times 1), \\
\mathbf{e}_{\mathrm{B}_2}(n) &= \chi(n-L) - \mathbf{B}^T(n-1)\tilde{\mathbf{x}}(n), \quad (P \times 1), \\
\mathbf{e}_{\mathrm{B}}(n) &= \kappa\mathbf{e}_{\mathrm{B}_2}(n) + (1-\kappa)\mathbf{e}_{\mathrm{B}_1}(n), \quad (P \times 1), \\
\mathbf{k}'(n) &= \mathbf{t}(n) + \mathbf{B}(n-1)\mathbf{m}(n), \quad (PL \times 1), \\
\varphi(n) &= \varphi_1(n) - \mathbf{e}_{\mathrm{B}_2}^T(n)\mathbf{m}(n), \quad (1 \times 1), \\
\mathbf{E}_{\mathrm{B}}(n) &= \lambda[\mathbf{E}_{\mathrm{B}}(n-1) + \mathbf{e}_{\mathrm{B}_2}(n)\mathbf{e}_{\mathrm{B}_2}^T(n)/\varphi(n)], \quad (P \times P), \\
\mathbf{B}(n) &= \mathbf{B}(n-1) + \mathbf{k}'(n)\mathbf{e}_{\mathrm{B}}^T(n)/\varphi(n), \quad (PL \times P). \\
& \quad \textit{Filtering}: \\
e_q(n) &= y_q(n) - \tilde{\mathbf{h}}_q^T(n-1)\tilde{\mathbf{x}}(n), \quad (1 \times 1), \\
\tilde{\mathbf{h}}_q(n) &= \tilde{\mathbf{h}}_q(n-1) + \mathbf{k}'(n)e_q(n)/\varphi(n), \quad (PL \times 1).
\end{aligned}$$

5. THE MULTICHANNEL LMS ALGORITHM

We derive two different versions of the multichannel LMS algorithm. The first one is straightforward and is a simple generalization of the single-channel LMS. The second one is more sophisticated and takes into account the cross-correlation among all the channels.

5.1 CLASSICAL DERIVATION

The mean-square error criterion is defined as

$$J_{\mathrm{MS},q} \;=\; E\left\{\left[y_q(n) - \mathbf{x}^T(n)\hat{\mathbf{h}}_q\right]^2\right\}, \tag{5.35}$$

where $E\{\cdot\}$ denotes mathematical expectation. Let $\mathbf{f}(\hat{\mathbf{h}}_q)$ denote the value of the gradient vector with respect to $\hat{\mathbf{h}}_q$. According to the steepest-descent method, the updated value of $\hat{\mathbf{h}}_q$ at time n is computed by using the simple recursive relation [17]:

$$\hat{\mathbf{h}}_q(n) \;=\; \hat{\mathbf{h}}_q(n-1) + \frac{\mu}{2}\left\{-\mathbf{f}\left[\hat{\mathbf{h}}_q(n-1)\right]\right\}, \tag{5.36}$$

where μ is positive step-size constant. Differentiating (5.35) with respect to the filter, we get the following value for the gradient vector:

$$\begin{aligned}\mathbf{f}(\hat{\mathbf{h}}_q) &= \left[\begin{array}{cccc}\mathbf{f}_1^T(\hat{\mathbf{h}}_q) & \mathbf{f}_2^T(\hat{\mathbf{h}}_q) & \cdots & \mathbf{f}_P^T(\hat{\mathbf{h}}_q)\end{array}\right]^T \\ &= \partial J_{\mathrm{MS},q}/\partial\hat{\mathbf{h}}_q = -2\mathbf{r}_{xy,q} + 2\mathbf{R}_{xx}\hat{\mathbf{h}}_q,\end{aligned} \tag{5.37}$$

with $\mathbf{r}_{xy,q} = E\{y_q(n)\mathbf{x}(n)\}$ and $\mathbf{R}_{xx} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$. By taking $\mathbf{f}(\hat{\mathbf{h}}_q) = \mathbf{0}_{LP\times 1}$, we obtain the Wiener-Hopf equations

$$\mathbf{R}_{xx}\hat{\mathbf{h}}_q = \mathbf{r}_{xy,q}, \tag{5.38}$$

which are similar to the normal equations (5.8) that were derived from a weighted least-squares criterion (5.4). Note that we use the same notation for similar variables that are derived either from the Wiener-Hopf equations or the normal equations.

The steepest-descent algorithm is now:

$$\hat{\mathbf{h}}_q(n) = \hat{\mathbf{h}}_q(n-1) + \mu E\{\mathbf{x}(n)e_q(n)\}, \tag{5.39}$$

and the classical stochastic approximation (consisting of approximating the gradient with its instantaneous value) [17] provides the multichannel LMS algorithm:

$$\hat{\mathbf{h}}_q(n) = \hat{\mathbf{h}}_q(n-1) + \mu\mathbf{x}(n)e_q(n), \tag{5.40}$$

of which the classical mean weight convergence condition under appropriate independence assumption is:

$$0 < \mu < \frac{2}{L\sum_{p=1}^{P}\sigma_{x_p}^2}, \tag{5.41}$$

where the $\sigma_{x_p}^2$ $(p = 1, 2, ..., P)$ are the powers of the input signals. When this condition is satisfied, the weight vector converges in the mean to the optimal Wiener-Hopf solution.

However, the gradient vector corresponding to the filter pq is:

$$\mathbf{f}_p(\hat{\mathbf{h}}_q) = -2\left(\mathbf{r}_{pq} - \sum_{j=1}^{P}\mathbf{R}_{pj}\hat{\mathbf{h}}_{jq}\right), \; p = 1, 2, ..., P, \tag{5.42}$$

which clearly shows some dependency of $\mathbf{f}_p$ on the full vector $\hat{\mathbf{h}}_q$. In other words, the filters $\hat{\mathbf{h}}_{jq}$ with $j \neq p$ influence, in a bad direction, the gradient vector $\mathbf{f}_p$ when seeking the minimum, because the algorithm does not take the cross-correlation among all the inputs into account.

5.2 IMPROVED VERSION

We have seen that during the convergence of the multichannel LMS algorithm, each adaptive filter depends of the others. This dependency must be taken into account. By using this information and Lemma 1, we now differentiate criterion (5.35) with respect to the tap-weight in a different way. The new gradient is obtained by writing that $\hat{\mathbf{h}}_{pq}$ depends of the full vector $\hat{\mathbf{h}}_q$. We get:

$$\begin{aligned} \mathbf{f}_p(\hat{\mathbf{h}}_{pq}) &= \frac{\partial J_{\mathrm{MS},q}}{\partial \hat{\mathbf{h}}_{pq}}(\hat{\mathbf{h}}_q) \qquad (5.43) \\ &= -2E\left\{\mathbf{z}_p(n)\left[y_q(n) - \mathbf{x}^T(n)\hat{\mathbf{h}}_q\right]\right\}, \; p = 1, 2, ..., P, \end{aligned}$$

with

$$\begin{aligned} \mathbf{z}_p(n) &= \sum_{j=1}^{P}[\partial\hat{\mathbf{h}}_{jq}/\partial\hat{\mathbf{h}}_{pq}]^T\mathbf{x}_j(n) \\ &= \sum_{j=1}^{P}\mathbf{C}_{pj}\mathbf{x}_j(n), \; p = 1, 2, ..., P. \end{aligned} \tag{5.44}$$

We have some interesting orthogonality and decorrelation properties.

Lemma 2:

$$\begin{aligned} E\{\mathbf{x}_p^T(n)\mathbf{z}_j(n)\} &= 0, & (5.45) \\ E\{\mathbf{z}_p(n)\mathbf{x}_j^T(n)\} &= \mathbf{0}_{L\times L}, \; \forall p, j = 1, 2, ..., P, \; p \neq j. & (5.46) \end{aligned}$$

Proof: The proof is straightforward from Lemma 1 (using mathematical expectation instead of weighted least-squares).

We can verify by using Lemma 2 that each gradient vector $\mathbf{f}_p$ $(p = 1, 2, ..., P)$ now depends only of the corresponding filter $\hat{\mathbf{h}}_{pq}$. In other words, we make the convergence of each $\hat{\mathbf{h}}_{pq}$ independent of the others, which is not the case in the classical gradient algorithm.

Based on the above gradient vector, the improved steepest-descent algorithm is easily obtained, out of which a stochastic approximation leads to the improved multichannel LMS algorithm:

$$\hat{\mathbf{h}}_q(n) = \hat{\mathbf{h}}_q(n-1) + \mu \mathbf{z}(n) e_q(n) \tag{5.47}$$

with

$$\mathbf{z}(n) = \left[\begin{array}{cccc} \mathbf{z}_1^T(n) & \mathbf{z}_2^T(n) & \cdots & \mathbf{z}_P^T(n) \end{array}\right]^T$$

and

$$0 < \mu < \frac{2}{L \sum_{p=1}^{P} \sigma_{z_p}^2} \tag{5.48}$$

to guaranty the convergence of the algorithm. Note that the improved multichannel LMS algorithm can be seen as an approximation of the factorized multichannel RLS algorithm by taking $\mathbf{R}_p^{-1}(n) \approx \mu \mathbf{I}_{L \times L}$.

6. THE MULTICHANNEL APA

The affine projection algorithm (APA) [18] has become popular because of its lower complexity compared to RLS while it converges almost as fast in the single-channel case. Therefore it is interesting to derive and study the multichannel version of this algorithm. Like the multichannel LMS, two versions are derived.

6.1 THE STRAIGHTFORWARD MULTICHANNEL APA

A simple trick for obtaining the single-channel APA is to search for an algorithm of the stochastic gradient type cancelling *N a posteriori* errors [19]. This requirement results in an underdetermined set of linear equations of which the mininum-norm solution is chosen. In the following, this technique is extended to the multichannel case [20].

By definition, the set of *N a priori* errors and *N a posteriori* errors are:

$$\begin{aligned} \mathbf{e}_q(n) &= \mathbf{y}_q(n) - \mathbf{X}^T(n)\hat{\mathbf{h}}_q(n-1), && (5.49)\\ \mathbf{e}_{\mathrm{a},q}(n) &= \mathbf{y}_q(n) - \mathbf{X}^T(n)\hat{\mathbf{h}}_q(n), && (5.50) \end{aligned}$$

where

$$\mathbf{X}(n) = \left[\ \mathbf{X}_1^T(n) \quad \mathbf{X}_2^T(n) \quad \cdots \quad \mathbf{X}_P^T(n)\ \right]^T$$

is a matrix of size $PL \times N$; the $L \times N$ matrix

$$\mathbf{X}_p(n) = \left[\ \mathbf{x}_p(n) \quad \mathbf{x}_p(n-1) \quad \cdots \quad \mathbf{x}_p(n-N+1)\ \right]$$

is made from the N last input vectors $\mathbf{x}_p(n)$; finally, $\mathbf{y}_q(n)$ and $\mathbf{e}_q(n)$ are respectively vectors of the N last samples of the reference signal $y_q(n)$ and error signal $e_q(n)$.

Using (5.49) and (5.50) plus the requirement that $\mathbf{e}_{\mathrm{a},q}(n) = \mathbf{0}_{N\times 1}$, weobtain:

$$\mathbf{X}^T(n)\Delta\hat{\mathbf{h}}_q(n) = \mathbf{e}_q(n), \tag{5.51}$$

where $\Delta\hat{\mathbf{h}}_q(n) = \hat{\mathbf{h}}_q(n) - \hat{\mathbf{h}}_q(n-1)$.

Equation (5.51) (N equations in PLunknowns, $N \leq PL$) is an underdetermined set of linear equations. Hence, it has an infinite number of solutions, out of which the minimum-norm solution is chosen. This results in [20], [21]:

$$\hat{\mathbf{h}}_q(n) = \hat{\mathbf{h}}_q(n-1) + \mathbf{X}(n)\left[\mathbf{X}^T(n)\mathbf{X}(n)\right]^{-1}\mathbf{e}_q(n). \tag{5.52}$$

However, in this straightforward APA, the normalization matrix $\mathbf{X}^T(n)\mathbf{X}(n) = \sum_{p=1}^{P}\mathbf{X}_p^T(n)\mathbf{X}_p(n)$ does not involve the cross-correlation elements of the P input signals [namely $\mathbf{X}_i^T(n)\mathbf{X}_p(n)$, $i, p = 1, 2, ..., P$, $i \neq p$] and this algorithm may converge slowly.

6.2 THE IMPROVED TWO-CHANNEL APA

A simple way to improve the previous adaptive algorithm is to use the othogonality and decorrelation properties, which will be shown later to appear in this context. Let us derive the improved algorithm by requiring a condition similar to the one used in the improved multichannel LMS. Just use the constraint that $\Delta\hat{\mathbf{h}}_{pq}$ be orthogonal to $\mathbf{X}_j$, $j \neq p$. As a result, we take into account separately the contributions of each input signal. These constraints read:

$$\begin{aligned} \mathbf{X}_2^T(n)\Delta\hat{\mathbf{h}}_{1q}(n) &= \mathbf{0}_{N\times 1}, & (5.53)\\ \mathbf{X}_1^T(n)\Delta\hat{\mathbf{h}}_{2q}(n) &= \mathbf{0}_{N\times 1}, & (5.54)\end{aligned}$$

and the new set of linear equations characterizing the improved two-channel APA is:

$$\begin{bmatrix} \mathbf{X}_1^T(n) & \mathbf{X}_2^T(n) \\ \mathbf{X}_2^T(n) & \mathbf{0}_{N\times L} \\ \mathbf{0}_{N\times L} & \mathbf{X}_1^T(n) \end{bmatrix} \begin{bmatrix} \Delta\hat{\mathbf{h}}_{1q}(n) \\ \Delta\hat{\mathbf{h}}_{2q}(n) \end{bmatrix} = \begin{bmatrix} \mathbf{e}_q(n) \\ \mathbf{0}_{N\times 1} \\ \mathbf{0}_{N\times 1} \end{bmatrix}. \tag{5.55}$$

The improved two-channel APA algorithm is given by the minimum-norm solution of (5.55) which is found as [20],

$$\Delta\hat{\mathbf{h}}_{1q}(n) = \mathbf{Z}_1(n)\left[\mathbf{Z}_1^T(n)\mathbf{Z}_1(n)+\mathbf{Z}_2^T(n)\mathbf{Z}_2(n)\right]^{-1}\mathbf{e}_q(n), \quad (5.56)$$

$$\Delta\hat{\mathbf{h}}_{2q}(n) = \mathbf{Z}_2(n)\left[\mathbf{Z}_1^T(n)\mathbf{Z}_1(n)+\mathbf{Z}_2^T(n)\mathbf{Z}_2(n)\right]^{-1}\mathbf{e}_q(n), \quad (5.57)$$

where $\mathbf{Z}_p(n)$ is the projection of $\mathbf{X}_p(n)$ onto a subspace orthogonal to $\mathbf{X}_j(n)$, $p \neq j$, i.e.,

$$\mathbf{Z}_p(n) = \left\{\mathbf{I}_{L\times L}-\mathbf{X}_j(n)\left[\mathbf{X}_j^T(n)\mathbf{X}_j(n)\right]^{-1}\mathbf{X}_j(n)\right\}\mathbf{X}_p(n), \quad (5.58)$$

$$p, j = 1, 2,\ p \neq j.$$

This results in the following orthogonality conditions,

$$\mathbf{X}_p^T(n)\mathbf{Z}_j(n) = \mathbf{0}_{N\times N},\ p\neq j \quad (5.59)$$

which are similar to what appears in the improved multichannel LMS (Lemma 2).

6.3 THE IMPROVED MULTICHANNEL APA

The algorithm explained for two channels is easily generalized to an arbitrary number of channels P. Define the following matrix of size $L \times (P-1)N$:

$$\underline{\mathbf{X}}_p(n) = \begin{bmatrix}\mathbf{X}_1(n) & \cdots & \mathbf{X}_{p-1}(n) & \mathbf{X}_{p+1}(n) & \cdots & \mathbf{X}_P(n)\end{bmatrix},$$

$$p = 1, 2, ..., P.$$

The P orthogonality constraints are:

$$\underline{\mathbf{X}}_p^T(n)\Delta\hat{\mathbf{h}}_{pq}(n) = \mathbf{0}_{(P-1)N\times 1},\ p = 1, 2, ..., P, \quad (5.60)$$

and by using the same steps as for $P = 2$, a solution similar to (5.56), (5.57) is obtained [20]:

$$\Delta\hat{\mathbf{h}}_{pq}(n) = \mathbf{Z}_p(n)\left[\sum_{j=1}^{P}\mathbf{Z}_j^T(n)\mathbf{Z}_j(n)\right]^{-1}\mathbf{e}_q(n),\ p = 1, 2, ..., P, \quad (5.61)$$

where $\mathbf{Z}_p(n)$ is the projection of $\mathbf{X}_p(n)$ onto a subspace orthogonal to $\underline{\mathbf{X}}_p(n)$, i.e.,

$$\mathbf{Z}_p(n) = \left\{\mathbf{I}_{L\times L}-\underline{\mathbf{X}}_p(n)\left[\underline{\mathbf{X}}_p^T(n)\underline{\mathbf{X}}_p(n)\right]^{-1}\underline{\mathbf{X}}_p(n)\right\}\mathbf{X}_p(n), \quad (5.62)$$

$$p = 1, 2, ..., P.$$

Note that this equation holds only under the condition $L \geq (P-1)N$, so that the matrix that appears in (5.62) be invertible.

We can easily see that:

$$\underline{\mathbf{X}}_p^T(n)\mathbf{Z}_p(n) = \mathbf{0}_{(P-1)N \times N},\ p = 1, 2, ..., P. \tag{5.63}$$

Fast versions of the single- and multi-channel APA can be derived [22], [23], [24].

7. THE MULTICHANNEL EXPONENTIATED GRADIENT ALGORITHM

Room acoustic impulse responses are often sparse. Our interest in exponentiated adaptive algorithms is that they converge and track much faster than the LMS algorithm for this family of impulse responses.

One easy way to find adaptive algorithms that adjust the new weight vector $\hat{\mathbf{h}}_q(n+1)$ from the old one $\hat{\mathbf{h}}_q(n)$ is to minimize the following function [25]:

$$J[\hat{\mathbf{h}}_q(n+1)] = d[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)] + \eta e_{\mathrm{a},q}^2(n+1), \tag{5.64}$$

where $d[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)]$ is some measure of distance from the old to the new weight vector,

$$e_{\mathrm{a},q}(n+1) = y_q(n+1) - \hat{\mathbf{h}}_q^T(n+1)\mathbf{x}(n+1) \tag{5.65}$$

is the *a posteriori* error signal, and η is a positive constant. (This formulation is a generalization of the case of Euclidean distance.) The magnitude of η represents the importance of correctness compared to the importance of conservativeness [25]. If η is very small, minimizing $J[\hat{\mathbf{h}}_q(n+1)]$ is close to minimizing $d[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)]$, so that the algorithm makes very small updates. On the other hand, if η is very large, the minimization of $J[\hat{\mathbf{h}}_q(n+1)]$ is almost equivalent to minimizing $d[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)]$ subject to the constraint $e_{\mathrm{a},q}(n+1) = 0$.

To minimize $J[\hat{\mathbf{h}}_q(n+1)]$, we need to set its *PL* partial derivatives $\partial J[\hat{\mathbf{h}}_q(n+1)]/\partial \hat{h}_{pq,l}(n+1)$ to zero. Hence, the different weight coefficients $\hat{h}_{pq,l}(n+1)$, $l = 0, 1, ..., L-1, p = 1, 2, ..., P$, will be found by solving the equations:

$$\frac{\partial d[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)]}{\partial \hat{h}_{pq,l}(n+1)} - 2\eta x_p(n+1-l)e_{\mathrm{a},q}(n+1) = 0. \tag{5.66}$$

Solving (5.66) is in general very difficult. However, if the new weight vector $\hat{\mathbf{h}}_q(n+1)$ is close to the old weight vector $\hat{\mathbf{h}}_q(n)$, replacing the *a posteriori* error signal $e_{\mathrm{a},q}(n+1)$ in (5.66) with the *a priori* error signal $e_q(n+1)$ is a reasonable approximation and the equation

$$\frac{\partial d[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)]}{\partial \hat{h}_{pq,l}(n+1)} - 2\eta x_p(n+1-l)e_q(n+1) = 0 \tag{5.67}$$

is much easier to solve for all distance measures d.

The LMS algorithm is easily obtained from (5.67) by using the squared Euclidean distance

$$d_{\mathrm{E}}[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)] = \|\hat{\mathbf{h}}_q(n+1) - \hat{\mathbf{h}}_q(n)\|_2^2. \tag{5.68}$$

The exponentiated gradient (EG) algorithm with positive weights results from using for d the *relative entropy,* also known as *Kullback-Leibler divergence,*

$$d_{\mathrm{re}}[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)] = \sum_{p=1}^{P} \sum_{l=0}^{L-1} \hat{h}_{pq,l}(n+1) \ln \frac{\hat{h}_{pq,l}(n+1)}{\hat{h}_{pq,l}(n)}, \tag{5.69}$$

with the constraint $\sum_p \sum_l \hat{h}_{pq,l}(n+1) = 1$, so that (5.67) becomes:

$$\frac{\partial d_{\mathrm{re}}[\hat{\mathbf{h}}_q(n+1), \hat{\mathbf{h}}_q(n)]}{\partial \hat{h}_{pq,l}(n+1)} - 2\eta x_p(n+1-l)e_q(n+1) + \gamma = 0, \tag{5.70}$$

where γ is the Lagrange multiplier. Actually, the appropriate constraint should be $\sum_p \sum_l \hat{h}_{pq,l}(n+1) = \sum_p \sum_l h_{pq,l}$ but $\sum_p \sum_l h_{pq,l}$ is not known in practice, so we use the arbitrary value 1 instead.

The algorithm derived from (5.70) is:

$$\hat{h}_{pq,l}(n+1) = \frac{\hat{h}_{pq,l}(n) r_{pq,l}(n+1)}{\sum_{i=1}^{P} \sum_{j=0}^{L-1} \hat{h}_{iq,j}(n) r_{iq,j}(n+1)}, \tag{5.71}$$

where

$$r_{pq,l}(n+1) = \exp\left[2\eta x_p(n+1-l)e_q(n+1)\right]. \tag{5.72}$$

This algorithm is valid only for positive coefficients. To deal with both positive and negative coefficients, we can always find two vectors $\hat{\mathbf{h}}_q^+(n+1)$ and $\hat{\mathbf{h}}_q^-(n+1)$ with positive coefficients, in such a way that the vector

$$\hat{\mathbf{h}}_q(n+1) = \hat{\mathbf{h}}_q^+(n+1) - \hat{\mathbf{h}}_q^-(n+1) \tag{5.73}$$

can have positive and negative components. In this case, the *a posteriori* error signal can be written as:

$$e_{\mathrm{a},q}(n+1) = y_q(n+1) - [\hat{\mathbf{h}}_q^+(n+1) - \hat{\mathbf{h}}_q^-(n+1)]^T \mathbf{x}(n+1) \tag{5.74}$$

and the function (5.64) will change to:

$$\begin{aligned} J[\hat{\mathbf{h}}_q^+(n+1), \hat{\mathbf{h}}_q^-(n+1)] &= d[\hat{\mathbf{h}}_q^+(n+1), \hat{\mathbf{h}}_q^+(n)] \\ &+ d[\hat{\mathbf{h}}_q^-(n+1), \hat{\mathbf{h}}_q^-(n)] + \frac{\eta}{u} e_{\mathrm{a},q}^2(n+1), \end{aligned} \tag{5.75}$$

where u is a positive scaling constant. Using the same approximation as before and choosing the Kullback-Leibler divergence plus the constraint $\sum_p \sum_l [\hat{h}_{pq,l}^+(n + 1) + \hat{h}_{pq,l}^-(n+1)] = u$, the solutions of the equations

$$\frac{\partial d_{\mathrm{re}}[\hat{\mathbf{h}}_q^+(n+1), \hat{\mathbf{h}}_q^+(n)]}{\partial \hat{h}_{pq,l}^+(n+1)} - 2\frac{\eta}{u} x_p(n+1-l) e_q(n+1) + \gamma = 0, \tag{5.76}$$

$$\frac{\partial d_{\mathrm{re}}[\hat{\mathbf{h}}_q^-(n+1), \hat{\mathbf{h}}_q^-(n)]}{\partial \hat{h}_{pq,l}^-(n+1)} + 2\frac{\eta}{u} x_p(n+1-l) e_q(n+1) + \gamma = 0, \tag{5.77}$$

give the so-called EG± algorithm:

$$\hat{h}_{pq,l}^+(n+1) = u \frac{\hat{h}_{pq,l}^+(n) r_{pq,l}^+(n+1)}{s_q(n+1)}, \tag{5.78}$$

$$\hat{h}_{pq,l}^-(n+1) = u \frac{\hat{h}_{pq,l}^-(n) r_{pq,l}^-(n+1)}{s_q(n+1)}, \tag{5.79}$$

where

$$s_q(n+1) = \sum_{i=1}^{P} \sum_{j=0}^{L-1} \left[\hat{h}_{iq,j}^+(n) r_{iq,j}^+(n+1) + \hat{h}_{iq,j}^-(n) r_{iq,j}^-(n+1) \right], \tag{5.80}$$

$$r_{pq,l}^+(n+1) = \exp\left[\frac{2\eta}{u} x_p(n+1-l) e_q(n+1) \right], \tag{5.81}$$

$$\begin{aligned} r_{pq,l}^-(n+1) &= \exp\left[-\frac{2\eta}{u} x_p(n+1-l) e_q(n+1) \right] \\ &= \frac{1}{r_{pq,l}^+(n+1)}, \end{aligned} \tag{5.82}$$

$$e_q(n+1) = y_q(n+1) - [\hat{\mathbf{h}}_q^+(n) - \hat{\mathbf{h}}_q^-(n)]^T \mathbf{x}(n+1). \tag{5.83}$$

We can check that we always have $\|\hat{\mathbf{h}}_q^+(n+1)\|_1 + \|\hat{\mathbf{h}}_q^-(n+1)\|_1 = u$. Upon convergence:

$$\begin{aligned}
\|\hat{\mathbf{h}}_q(\infty)\|_1 &= \|\mathbf{h}_q\|_1 \\
&= \|\hat{\mathbf{h}}_q^+(\infty) - \hat{\mathbf{h}}_q^-(\infty)\|_1 \\
&\leq \|\hat{\mathbf{h}}_q^+(\infty)\|_1 + \|\hat{\mathbf{h}}_q^-(\infty)\|_1 = u, \qquad (5.84)
\end{aligned}$$

hence, the constant u should be chosen such that $u \geq \|\mathbf{h}_q\|_1$.

A normalized version of the multichannel EG± algorithm is given below:

$$\begin{aligned}
& \textit{Initialization}: \\
\hat{h}_{pq,l}^+(0) &= \hat{h}_{pq,l}^-(0) = c > 0,\ p = 1, 2, ..., P,\ l = 0, 1, ..., L-1. \\
& \textit{Parameters}: \\
u &\geq \|\mathbf{h}_q\|_1, \\
0 < \alpha &\leq 1,\ \delta > 0. \\
& \textit{Error}: \\
e_q(n+1) &= y_q(n+1) - [\hat{\mathbf{h}}_q^+(n) - \hat{\mathbf{h}}_q^-(n)]^T \mathbf{x}(n+1). \\
& \textit{Update}: \\
\mu(n+1) &= \frac{\alpha}{\mathbf{x}^T(n+1)\mathbf{x}(n+1) + \delta}, \\
r_{pq,l}^+(n+1) &= \exp\left[PL\frac{\mu(n+1)}{u} x_p(n+1-l) e_q(n+1)\right], \\
r_{pq,l}^-(n+1) &= \frac{1}{r_{pq,l}^+(n+1)}, \\
s_q(n+1) &= \sum_{i=1}^{P} \sum_{j=0}^{L-1} \left[\hat{h}_{iq,j}^+(n) r_{iq,j}^+(n+1) + \hat{h}_{iq,j}^-(n) r_{iq,j}^-(n+1)\right], \\
\hat{h}_{pq,l}^+(n+1) &= u\frac{\hat{h}_{pq,l}^+(n) r_{pq,l}^+(n+1)}{s_q(n+1)}, \\
\hat{h}_{pq,l}^-(n+1) &= u\frac{\hat{h}_{pq,l}^-(n) r_{pq,l}^-(n+1)}{s_q(n+1)}, \\
& p = 1, 2, ..., P,\ l = 0, 1, ..., L-1.
\end{aligned}$$

Intuitively, exponentiating the update has the effect of assigning larger relative updates to larger weights, thereby deemphasizing the effect of smaller weights. This is qualitatively similar to the PNLMS algorithm [26] which makes the update *proportional* to the size of the weight. This type of behavior is desirable for sparse impulse responses where small weights do not contribute significantly to the *mean* solution but introduce an undesirable noise-like *variance.*

Recently, the proportionate normalized least-mean-square (PNLMS) algorithm was developed for use in network echo cancelers [26]. In comparison to the NLMS algorithm, PNLMS has very fast initial convergence and tracking when the echo path is sparse. As previously mentioned, the idea behind PNLMS is to update each coefficient of the filter independently of the others by adjusting the adaptation step size in proportion to the estimated filter coefficient. More recently, an improved PNLMS (IPNLMS) [27] was proposed that performs better than NLMS and PNLMS, whatever the nature of the impulse response is. The IPNLMS is summarized below:

$$\begin{aligned}
& && \textit{Initialization}: \\
\hat{h}_{pq,l}(0) &=& & 0,\ p = 1, 2, ..., P,\ l = 0, 1, ..., L-1. \\
& && \textit{Parameters}: \\
0 < &\ \alpha & & \leq 1,\ \delta_{\text{IPNLMS}} > 0, \\
-1 &\ \leq & & \kappa \leq 1, \\
&\ \varepsilon & & > 0 \text{ (small number to avoid division by zero).} \\
& && \textit{Error}: \\
e_q(n+1) &=& & y_q(n+1) - \hat{\mathbf{h}}_q^T(n)\mathbf{x}(n+1). \\
& && \textit{Update}: \\
g_{pq,l}(n) &=& & \frac{1-\kappa}{2PL} + (1+\kappa)\frac{|\hat{h}_{pq,l}(n)|}{2\|\hat{\mathbf{h}}_q(n)\|_1 + \varepsilon}, \\
p &=& & 1, 2, ..., P,\ l = 0, 1, ..., L-1, \\
\mu(n+1) &=& & \frac{\alpha}{\sum_{i=1}^{P}\sum_{j=0}^{L-1} x_i^2(n+1-j)g_{iq,j}(n) + \delta_{\text{IPNLMS}}}, \\
\hat{h}_{pq,l}(n+1) &=& & \hat{h}_{pq,l}(n) + \mu(n+1)g_{pq,l}(n)x_p(n+1-l)e_q(n+1), \\
p &=& & 1, 2, ..., P,\ l = 0, 1, ..., L-1.
\end{aligned}$$

In general, $g_{pq,l}$ in the IPNLMS provides the "proportionate" scaling of the update. The parameter κ controls the amount of proportionality in the update. For $\kappa = -1$, it can easily be checked that the IPNLMS and NLMS algorithms are identical. For κ close to 1, the IPNLMS behaves like the PNLMS algorithm [26]. In practice, a good choice for κ is 0 or –0.5.

We can show that the IPNLMS and EG± algorithms are related [28]. The IPNLMS is in fact an approximation of the EG± if we approximate in the latter $\exp(a) \approx 1 + a$ for $|a| \ll 1$.

8. THE MULTICHANNEL FREQUENCY-DOMAIN ADAPTIVE ALGORITHM

Adaptive algorithms in the frequency domain are, in general, extremely efficient since they use the fast Fourier transform (FFT) as an intermediary step. As a result, they are now implemented in many prototypes and products for acoustic echo cancellation. In this section, we briefly explain how these algorithms can be derived rigorously from a block error signal.

From now on and for simplification, we drop the parameter q in all equations. With this simplification, the error signal at time n is now:

$$e(n) = y(n) - \sum_{p=1}^{P} \hat{\mathbf{h}}_p^T \mathbf{x}_p(n), \tag{5.85}$$

where $\hat{\mathbf{h}}_p$ is the estimated impulse response of the pth channel,

$$\hat{\mathbf{h}}_p = \left[\begin{array}{cccc} \hat{h}_{p,0} & \hat{h}_{p,1} & \cdots & \hat{h}_{p,L-1} \end{array} \right]^T .$$

We now define the block error signal (of length $N \leq L$). For that, we assume that L is an integer multiple of N, i.e., L = KN. We have:

$$\begin{aligned} \mathbf{e}(m) &= \mathbf{y}(m) - \hat{\mathbf{y}}(m) \\ &= \mathbf{y}(m) - \sum_{p=1}^{P} \mathbf{X}_p^T(m)\hat{\mathbf{h}}_p, \end{aligned} \tag{5.86}$$

where m is the block time index, and

$$\begin{aligned} \mathbf{e}(m) &= \left[\begin{array}{ccc} e(mN) & \cdots & e(mN+N-1) \end{array} \right]^T, \\ \mathbf{y}(m) &= \left[\begin{array}{ccc} y(mN) & \cdots & y(mN+N-1) \end{array} \right]^T, \\ \mathbf{X}_p(m) &= \left[\begin{array}{ccc} \mathbf{x}_p(mN) & \cdots & \mathbf{x}_p(mN+N-1) \end{array} \right], \\ \hat{\mathbf{y}}(m) &= \left[\begin{array}{ccc} \hat{y}(mN) & \cdots & \hat{y}(mN+N-1) \end{array} \right]^T. \end{aligned}$$

It can easily be checked that $\mathbf{X}_p$ is a Toeplitz matrix of size (L × N).

We can show that for K = L/N, we can write

$$\mathbf{X}_p^T(m)\hat{\mathbf{h}}_p = \sum_{k=0}^{K-1} \mathbf{T}_p(m-k)\hat{\mathbf{h}}_{p,k}, \tag{5.87}$$

where $\mathbf{T}(m-k)$ is an (N × N) Toeplitz matrix and

$$\hat{\mathbf{h}}_{p,k} = \left[\begin{array}{cccc} \hat{h}_{p,kN} & \hat{h}_{p,kN+1} & \cdots & \hat{h}_{p,kN+N-1} \end{array} \right]^T, \; k = 0, 1, ..., K-1,$$

are the sub-filters of $\hat{\mathbf{h}}_p$. In (5.87), the filter $\hat{\mathbf{h}}_p$ (of length L) is partitioned into K sub-filters $\hat{\mathbf{h}}_{p,k}$ of length N and the rectangular matrix $\mathbf{X}_p^T$ [of size $(N \times L)$] is decomposed to K square sub-matrices of size $(N \times N)$.

It is well known that a Toeplitz matrix $\mathbf{T}_p$ can be transformed, by doubling its size, to a circulant matrix $\mathbf{C}_p$. Also, a circulant matrix is easily decomposed as follows: $\mathbf{C}_p = \mathbf{F}_{2N\times 2N}^{-1}\mathbf{D}_p\mathbf{F}_{2N\times 2N}$, where $\mathbf{F}_{2N\times 2N}$ is the Fourier matrix [of size $(2N \times 2N)$] and $\mathbf{D}_p$ is a diagonal matrix whose elements are the discrete Fourier transform of the first column of $\mathbf{C}_p$. If we multiply (5.86) by $\mathbf{F}_{N\times N}$ [Fourier matrix of size $(N \times N)$], we get the error signal in the frequency domain (denoted by underbars):

$$\begin{aligned}
\underline{\mathbf{e}}(m) &= \underline{\mathbf{y}}(m) - \mathbf{G}_{N\times 2N}^{01}\sum_{p=1}^{P}\sum_{k=0}^{K-1}\mathbf{D}_p(m-k)\mathbf{G}_{2N\times N}^{10}\underline{\hat{\mathbf{h}}}_{p,k} \\
&= \underline{\mathbf{y}}(m) - \mathbf{G}_{N\times 2N}^{01}\sum_{p=1}^{P}\sum_{k=0}^{K-1}\mathbf{U}_p(m-k)\underline{\hat{\mathbf{h}}}_{p,k} \\
&= \underline{\mathbf{y}}(m) - \mathbf{G}_{N\times 2N}^{01}\sum_{p=1}^{P}\underline{\mathbf{U}}_p(m)\underline{\hat{\mathbf{h}}}_p \\
&= \underline{\mathbf{y}}(m) - \mathbf{G}_{N\times 2N}^{01}\underline{\mathbf{U}}(m)\underline{\hat{\mathbf{h}}}, \qquad (5.88)
\end{aligned}$$

where

$$\begin{aligned}
\underline{\mathbf{e}}(m) &= \mathbf{F}_{N\times N}\mathbf{e}(m), \\
\underline{\mathbf{y}}(m) &= \mathbf{F}_{N\times N}\mathbf{y}(m), \\
\mathbf{G}_{N\times 2N}^{01} &= \mathbf{F}_{N\times N}\mathbf{W}_{N\times 2N}^{01}\mathbf{F}_{2N\times 2N}^{-1}, \\
\mathbf{W}_{N\times 2N}^{01} &= \begin{bmatrix} \mathbf{0}_{N\times N} & \mathbf{I}_{N\times N} \end{bmatrix}, \\
\mathbf{G}_{2N\times N}^{10} &= \mathbf{F}_{2N\times 2N}\mathbf{W}_{2N\times N}^{10}\mathbf{F}_{N\times N}^{-1}, \\
\mathbf{W}_{2N\times N}^{10} &= \begin{bmatrix} \mathbf{I}_{N\times N} \\ \mathbf{0}_{N\times N} \end{bmatrix}, \\
\underline{\hat{\mathbf{h}}}_{p,k} &= \mathbf{F}_{N\times N}\hat{\mathbf{h}}_{p,k}, \\
\underline{\hat{\mathbf{h}}}_p &= \begin{bmatrix} \underline{\hat{\mathbf{h}}}_{p,0}^T & \underline{\hat{\mathbf{h}}}_{p,1}^T & \cdots & \underline{\hat{\mathbf{h}}}_{p,K-1}^T \end{bmatrix}^T, \\
\underline{\hat{\mathbf{h}}} &= \begin{bmatrix} \underline{\hat{\mathbf{h}}}_1^T & \underline{\hat{\mathbf{h}}}_2^T & \cdots & \underline{\hat{\mathbf{h}}}_P^T \end{bmatrix}^T, \\
\mathbf{D}_p(m-k) &= \mathbf{F}_{2N\times 2N}\mathbf{C}_p(m-k)\mathbf{F}_{2N\times 2N}^{-1}, \\
\mathbf{U}_p(m-k) &= \mathbf{D}_p(m-k)\mathbf{G}_{2N\times N}^{10}, \\
\underline{\mathbf{U}}_p(m) &= \begin{bmatrix} \mathbf{U}_p(m) & \mathbf{U}_p(m-1) & \cdots & \mathbf{U}_p(m-K+1) \end{bmatrix},
\end{aligned}$$

$$\begin{aligned}
\underline{\mathbf{U}}(m) &= [\ \underline{\mathbf{U}}_1(m)\ \ \underline{\mathbf{U}}_2(m)\ \ \cdots\ \ \underline{\mathbf{U}}_P(m)\ ], \\
\underline{\mathbf{D}}_p(m) &= [\ \mathbf{D}_p(m)\ \ \mathbf{D}_p(m-1)\ \ \cdots\ \ \mathbf{D}_p(m-K+1)\ ], \\
\underline{\mathbf{D}}(m) &= [\ \underline{\mathbf{D}}_1(m)\ \ \underline{\mathbf{D}}_2(m)\ \ \cdots\ \ \underline{\mathbf{D}}_P(m)\ ], \\
\mathbf{G}^{10}_{2PL\times PL} &= \mathrm{diag}[\ \mathbf{G}^{10}_{2N\times N}\ \ \cdots\ \ \mathbf{G}^{10}_{2N\times N}\ ], \\
\underline{\mathbf{U}}(m) &= \underline{\mathbf{D}}(m)\mathbf{G}^{10}_{2PL\times PL}.
\end{aligned}$$

The size of the matrix $\underline{\mathbf{U}}$ is $(2N \times PL)$ and the length of $\hat{\underline{\mathbf{h}}}$ is PL.

By minimizing the criterion

$$J_{\mathrm{f}}(m) = (1-\lambda)\sum_{i=0}^{m} \lambda^{m-i}\underline{\mathbf{e}}^H(i)\underline{\mathbf{e}}(i)\ , \tag{5.89}$$

where ^{H} denotes conjugate transpose and λ $(0 < \lambda < 1)$ is an exponential forgetting factor, we obtain the normal equations for the multichannel case:

$$\mathbf{S}(m)\hat{\underline{\mathbf{h}}}(m) = \mathbf{s}(m), \tag{5.90}$$

where

$$\begin{aligned}
\mathbf{S}(m) &= \lambda\mathbf{S}(m-1) \\
&+ (1-\lambda)(\mathbf{G}^{10}_{2PL\times PL})^H\underline{\mathbf{D}}^H(m)\mathbf{G}^{01}_{2N\times 2N}\underline{\mathbf{D}}(m)\mathbf{G}^{10}_{2PL\times PL}
\end{aligned} \tag{5.91}$$

is a $(PL \times PL)$ matrix,

$$\begin{aligned}
\mathbf{G}^{01}_{2N\times 2N} &= (\mathbf{G}^{01}_{N\times 2N})^H\mathbf{G}^{01}_{N\times 2N} \\
&= \mathbf{F}_{2N\times 2N}\mathbf{W}^{01}_{2N\times 2N}\mathbf{F}^{-1}_{2N\times 2N},
\end{aligned}$$

and

$$\mathbf{s}(m) = \lambda\mathbf{s}(m-1) + (1-\lambda)(\mathbf{G}^{10}_{2PL\times PL})^H\underline{\mathbf{D}}^H(m)\underline{\mathbf{y}}_{2N}(m) \tag{5.92}$$

is a $(PL \times 1)$ vector. Assuming that the P input signals are not perfectly pairwise coherent, the normal equations have a unique solution which is the optimal Wiener solution.

Define the following variables:

$$\begin{aligned}
\underline{\mathbf{y}}_{2N}(m) &= (\mathbf{G}^{01}_{N\times 2N})^H\underline{\mathbf{y}}(m) \\
&= \mathbf{F}_{2N\times 2N}\begin{bmatrix}\mathbf{0}_{N\times 1} \\ \mathbf{y}(m)\end{bmatrix}, \\
\underline{\mathbf{e}}_{2N}(m) &= \mathbf{F}_{2N\times 2N}\begin{bmatrix}\mathbf{0}_{N\times 1} \\ \mathbf{e}(m)\end{bmatrix} \\
&= (\mathbf{G}^{01}_{N\times 2N})^H\underline{\mathbf{e}}(m), \\
\mathbf{G}^{10}_{2PL\times 2PL} &= \mathrm{diag}[\ \mathbf{G}^{10}_{2N\times 2N}\ \ \cdots\ \ \mathbf{G}^{10}_{2N\times 2N}\ ], \\
\hat{\underline{\mathbf{h}}}_{2PL}(m) &= \mathbf{G}^{10}_{2PL\times PL}\hat{\underline{\mathbf{h}}}(m).
\end{aligned}$$

We can show that from the normal equations, we can exactly derive the following multichannel frequency-domain adaptive algorithm:

$$\begin{aligned}
\mathbf{Q}(m) &= \lambda\mathbf{Q}(m-1) + (1-\lambda)\underline{\mathbf{D}}^H(m)\mathbf{G}^{01}_{2N\times 2N}\underline{\mathbf{D}}(m), && (5.93)\\
\underline{\mathbf{e}}_{2N}(m) &= \underline{\mathbf{y}}_{2N}(m) - \mathbf{G}^{01}_{2N\times 2N}\underline{\mathbf{D}}(m)\hat{\underline{\mathbf{h}}}_{2PL}(m-1), && (5.94)\\
\hat{\underline{\mathbf{h}}}_{2PL}(m) &= \hat{\underline{\mathbf{h}}}_{2PL}(m-1) \\
&+ (1-\lambda)\mathbf{G}^{10}_{2PL\times 2PL}\mathbf{Q}^{-1}(m)\underline{\mathbf{D}}^H(m)\underline{\mathbf{e}}_{2N}(m). && (5.95)
\end{aligned}$$

Depending of the approximations we make on matrix $\mathbf{Q}(m)$, we obtain different algorithms. There is a compromise between performance (convergence rate) and complexity. Several algorithms can be deduced from the previous form, some of them are well-known while others are new. See [29] for a general derivation of adaptive algorithms in the frequency domain. See also [30] and [31] for new algorithms derived directly from the above equations.

9. CONCLUSIONS

In this chapter, we have given an overview on multichannel adaptive algorithms in the context of multichannel acoustic echo cancellation. We have first derived the normal equations of a MIMO system and discussed the identification problem. We have shown that, in the multichannel case and when the input signals are linearly related, there is a nonuniqueness problem that does not exist in the single-input single-output (SISO) case. In order to have a unique solution, we have to decorrelate somehow the input signals without affecting the spatialization effect and the quality of the signals. We have then derived many useful adaptive algorithms in a context where the strong correlation between the input signals affects seriously the performance of several well-known algorithms.

References

[1] M. M. Sondhi, D. R. Morgan, and J. L. Hall, "Stereophonic acoustic echo cancellation—An overview of the fundamental problem," *IEEE Signal Processing Lett.,* vol. 2, pp. 148–151, Aug. 1995.

[2] J. Benesty, D. R. Morgan, and M. M. Sondhi, "A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation," *IEEE Trans. Speech Audio Processing,* vol. 6, pp. 156–165, Mar. 1998.

[3] T. Gänsler and J. Benesty, "New insights into the stereophonic acoustic echo cancellation problem and an adaptive nonlinearity solution," *IEEE Trans. Speech Audio Processing,* vol. 10, pp. 257–267, July 2002.

[4] T. Gänsler and P. Eneroth, "Influence of audio coding on stereophonic acoustic echo cancellation," in *Proc. IEEE ICASSP,* 1998, pp. 3649–3652.

[5] A. Gilloire and V. Turbin, "Using auditory properties to improve the behavior of stereophonic acoustic echo cancellers," in *Proc. IEEE ICASSP,* 1998, pp. 3681–3684.

[6] S. Shimauchi, Y. Haneda, S. Makino, and Y. Kaneda, "New configuration for a stereo echo canceller with nonlinear pre-processing," in *Proc. IEEE ICASSP,* 1998, pp. 3685–3688.

[7] M. Ali, "Stereophonic echo cancellation system using time-varying all-pass filtering for signal decorrelation," in *Proc. IEEE ICASSP,* 1998, pp. 3689–3692.

[8] Y. Joncour and A. Sugiyama, "A stereo echo canceller with pre-processing for correct echo path identification," in *Proc. IEEE ICASSP,* 1998, pp. 3677–3680.

[9] J. Benesty, T. Gänsler, and P. Eneroth, "Multi-channel sound, acoustic echo cancellation, and multi-channel time-domain adaptive filtering," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., Kluwer Academic Publishers, Boston, 2000, Chap. 6.

[10] T. Gänsler and J. Benesty, "Stereophonic acoustic echo cancellation and two-channel adaptive filtering: an overview," *Int. J. Adapt. Control Signal Process.,* vol. 14, pp. 565–586, Sept. 2000.

[11] B. C. J. Moore, *An introduction to the Psychology of Hearing.* Academic Press, London, 1989, Chap. 3.

[12] D. R. Morgan, J. L. Hall, and J. Benesty, "Investigation of several types of nonlinearities for use in stereo acoustic echo cancellation," *IEEE Trans. Speech Audio Processing,* vol. 9, pp. 686–696, Sept. 2001.

[13] J. Benesty, D. R. Morgan, J. L. Hall, and M. M. Sondhi, "Stereophonic acoustic echo cancellation using nonlinear transformations and comb filtering," in *Proc. IEEE ICASSP,* 1998, pp. 3673–3676.

[14] J. Benesty, D. R. Morgan, and M. M. Sondhi, "A hybrid mono/stereo acoustic echo canceler," *IEEE Trans. Speech Audio Processing,* vol. 6, pp. 468–475, Sept. 1998.

[15] J. Benesty, F. Amand, A. Gilloire, and Y. Grenier, "Adaptive filtering algorithms for stereophonic acoustic echo cancellation," in *Proc. IEEE ICASSP,* 1995, pp. 3099–3102.

[16] M. G. Bellanger, *Adaptive Digital Filters and Signal Analysis.* Marcel Dekker, New York, 1987.

[17] S. Haykin, *Adaptive Filter Theory.* Fourth Edition, Prentice Hall, Upper Saddle River, N.J., 2002.

[18] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Elec. Comm. Japan,* vol. J67-A, pp. 126–132, Feb. 1984.

[19] M. Montazeri and P. Duhamel, "A set of algorithms linking NLMS and block RLS algorithms," *IEEE Trans. Signal Processing,* vol. 43, pp. 444–453, Feb. 1995.

[20] J. Benesty, P. Duhamel, and Y. Grenier, "A multi-channel affine projection algorithm with applications to multi-channel acoustic echo cancellation," *IEEE Signal Processing Lett.,* vol. 3, pp. 35–37, Feb. 1996.

[21] S. Shimauchi and S. Makino, "Stereo projection echo canceller with true echo path estimation," in *Proc. IEEE ICASSP,* 1995, pp. 3059–3062.

[22] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," in *Proc. IEEE ICASSP,* 1995, pp. 3023–3026.

[23] M. Tanaka, Y. Kaneda, S. Makino, and J. Kojima "Fast projection algorithm and its step size control," in *Proc. IEEE ICASSP,* 1995, pp. 945–948.

[24] F. Amand, J. Benesty, A. Gilloire, and Y. Grenier, "A fast two-channel affine projection algorithm for stereophonic acoustic echo cancellation," in *Proc. IEEE ICASSP,* 1996, pp. 949–952.

[25] J. Kivinen and M. K. Warmuth, “Exponentiated gradient versus gradient descent for linear predictors,” *Inform. Comput.,* vol. 132, pp. 1–64, Jan. 1997.

[26] D. L. Duttweiler, “Proportionate normalized least mean square adaptation in echo cancelers,” *IEEE Trans. Speech Audio Processing,* vol. 8, pp. 508–518, Sept. 2000.

[27] J. Benesty and S. L. Gay, “An improved PNLMS algorithm,” in *Proc. IEEE ICASSP,* 2002, pp. 1881–1884.

[28] J. Benesty, Y. Huang, and D. R. Morgan, “On a class of exponentiated adaptive algorithms for the identification of sparse impulse responses,” in *Adaptive Signal Processing—Applications to Real-World Problems,* J. Benesty and Y. Huang, Eds., Springer-Verlag, Berlin, 2003, Chap. 1.

[29] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo cancellation.* Springer-Verlag, Berlin, 2001.

[30] H. Buchner, J. Benesty, and W. Kellermann, “Multichannel frequency-domain adaptive filtering with application to multichannel acoustic echo cancellation,” in *Adaptive Signal Processing—Applications to Real-World Problems,* J. Benesty and Y. Huang, Eds., Springer-Verlag, Berlin, 2003, Chap. 4.

[31] H. Buchner, J. Benesty, and W. Kellermann, “An extended multidelay filter: fast low-delay algorithms for very high-order adaptive systems,” in *Proc. IEEE ICASSP,* 2003, pp. V-385–V-388.

This page intentionally left blank

Chapter 6

DOUBLE-TALK DETECTORS FOR ACOUSTIC ECHO CANCELERS

Tomas Gänsler
Agere Systems
gaensler@agere.com

Jacob Benesty
Université du Québec, INRS-EMT
benesty@inrs-emt.uquebec.ca

Abstract Double-talk detectors (DTDs) are vital to the operation and performance of acoustic echo cancelers. In this chapter, we highlight important aspects needed to be considered when choosing and designing a DTD. The generic double-talk detector scheme along with fundamental means of performance evaluation are discussed. A number of double-talk detectors suitable for acoustic echo cancelers are presented and objectively compared.

Keywords: Double-Talk Detector, Acoustic Echo Canceler, Adaptive Algorithms, LMS, NLMS, Coherence, Cross-Correlation, Robust Statistics

1. INTRODUCTION

> *The design of a good double-talk detector is much more of an art than the design of the adaptive filter itself.* [1][1]

Ideally, acoustic echo cancelers (AECs) remove undesired echoes that result from coupling between the loudspeaker and the microphone used in full-duplex hands-free telecommunication systems. Figure 6.1 shows a basic AEC block diagram. The far-end speech signal $x(n)$ goes through the echo path represented by a filter $h(n)$, then it is picked up by the microphone together with the near-end talker signal $v(n)$ and ambient noise $w(n)$. The microphone signal is denoted $y(n)$. Most often the echo path is modeled by an adaptive FIR filter, $\hat{h}(n)$, that generates a replica of the echo. This echo estimate is then subtracted

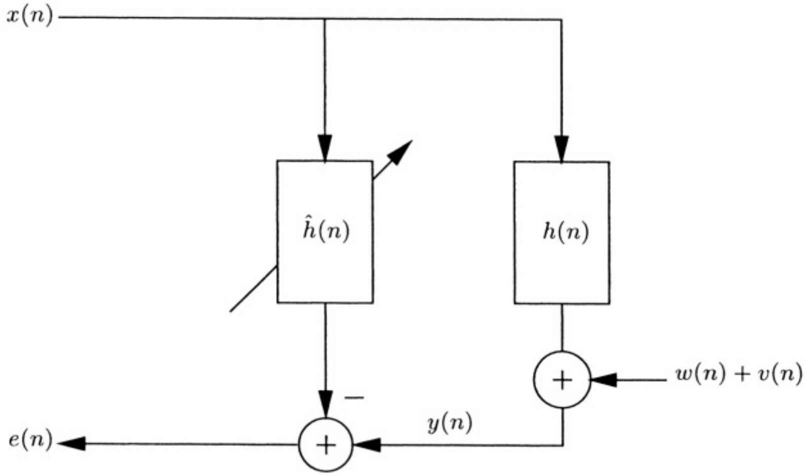


Figure 6.1 Block diagram of a basic AEC setup.

from the return channel and thereby cancellation is achieved. This may look like a simple straightforward system identification task for the adaptive filter. However, in most conversation there are so called *double-talk* situations that make the identification much more problematic than what it might appear at a first glance. Double-talk occurs when the speech of the two talkers arrive simultaneously at the echo canceler, i.e. $x(n) \neq 0$ and $v(n) \neq 0$ (the situation with near-end talk only, $x(n) = 0$ and $v(n) \neq 0$, can be regarded as an "easy-to-detect" double-talk case). In the double-talk situation, the near-end speech acts as a large level uncorrelated noise to the adaptive algorithm. The disturbing near-end speech may cause the adaptive filter to diverge. Hence, annoying audible echo will pass through to the far-end. The usual way to alleviate this problem is to slow down or completely halt the filter adaptation when presence of near-end speech is detected. This is the very important role of the so called double-talk detector (DTD). The basic double-talk detection scheme is based on computing a detection statistic, ξ, and comparing it with a preset threshold, *T*. The important issues that have to be addressed when designing a DTD are:

(i) What basic knowledge is needed in order to devise a sufficient DTD solution?

(ii) What characterize "good" double-talk detectors?

Primarily, we must know under what circumstances double-talk disturbs the adaptive filter. The performance of AECs are most often evaluated through their mean-square error (MSE) performance or preferably, through their misalignment $\varepsilon = \|\mathbf{h}-\hat{\mathbf{h}}\|_2/\|\mathbf{h}\|_2$ in different situations. The misalignment formula reveals the answer to question (i). This formula is given by

$$E\{\varepsilon^2\} \approx \frac{\mu_0}{2\mathrm{EBR}}, \tag{6.1}$$

where $E\{\cdot\}$ denotes mathematical expectation and μ_0 is a constant parameter of the adaptive algorithm. That is, the level of convergence (misalignment performance) is completely governed by *echo-to-background (near-end speech and noise)* power ratio, EBR. If we have equally strong talkers during double-talk, we find that when the echo path has a high attenuation, the adaptive filter will be very sensitive to double-talk. On the contrary, a low attenuation or amplification of the echo path means lower sensitivity. This is important to remember when chosing the parameters of the DTD. Furthermore, it naturally shows why we should lower the step-size parameter μ_0 in high noise or double-talk conditions in order to maintain a certain performance. Equation (6.1) is easily found by assuming the far- and near-end signals are uncorrelated stochastic processes and the adaptive algorithm is NLMS [3], Furthermore, if we suppose that the signals are white noises, the misalignment is exactly the inverse of the *echo return loss enhancement* (ERLE) which reflects the echo attenuation provided by the AEC. Even though these assumptions describe an oversimplified model of the AEC situation, it gives the valuable insight needed to what governs divergence of the adaptive algorithm.

Issue (ii) can be partially addressed by characterizing DTDs through the use of general detection theory [4, 5, 6]. By means of detection probability and false alarm rates we can objectively evaluate and compare the performance of different DTDs. Moreover, the theory justifies a method to select the threshold T which has been missing in the field of DTD design. One must, however, also accompany these performance measures with a joint evaluation of the DTD and the echo canceler.

A large number of DTD schemes have been proposed since the introduction of echo cancelers [7]. The Geigel algorithm [8] has proven successful in network echo cancelers; however, it does not always provide reliable performance when used in the acoustic situation. This is because it assumes a minimum echo path attenuation which may not be valid in the acoustic case. Other methods based on cross-correlation and coherence [9, 10,4, 5] have been studied which appear to be more appropriate for the acoustic application. Spectral comparing methods [11] and two-microphone solutions have also been proposed [12]. A DTD based on multi statistic testing in combination with modeling of the echo path by two filters is proposed in [13]. The objective of this chapter is to summarize some of the DTD proposals and present evaluation methods. Many results in this chapter are derived from the papers [5, 6].

The chapter is organized as follows: Section 2 introduces the AEC notations and describes the general DTD scheme. A number of double-talk detection algorithms that have been proposed and used in acoustic echo cancelers are presented in Section 3. Section 4 compares a selected number of DTDs by means of their receiver operation characteristics. Section 5 gives a discussion of the abilities of different DTD schemes and summarizes the important aspects that need to be considered for a successful double-talk detector implementation.

2. BASICS OF AEC AND DTD

In this section, we give the basics of an AEC combined with a DTD. We first formulate the AEC problem.

2.1 AEC NOTATIONS

The AEC setup in Fig. 6.1 is described in mathematical terms as:

$$y(n) = \mathbf{h}^T\mathbf{x}(n) + v(n) + w(n), \tag{6.2}$$

where

$$\mathbf{h} = \begin{bmatrix} h_0 & h_1 & \cdots & h_{N-1} \end{bmatrix}^T,$$
$$\mathbf{x}(n) = \begin{bmatrix} x(n) & x(n-1) & \cdots & x(n-N+1) \end{bmatrix}^T,$$

and *N* is the length of the echo path response, **h**. The error signal is defined as

$$\begin{aligned} e(n) &= y(n) - [\ \hat{\mathbf{h}}^T \quad \mathbf{0}^T\ ]\mathbf{x}(n) \\ &= \Delta\mathbf{h}^T\mathbf{x}(n) + v(n) + w(n), \end{aligned} \tag{6.3}$$

where

$$\hat{\mathbf{h}} = \begin{bmatrix} \hat{h}_0 & \hat{h}_1 & \cdots & \hat{h}_{L-1} \end{bmatrix}^T \tag{6.4}$$

is the adaptive filter coefficient vector of length *L* (generally less than *N),* and

$$\Delta\mathbf{h} = \mathbf{h} - \begin{bmatrix} \hat{\mathbf{h}} \\ \mathbf{0} \end{bmatrix}. \tag{6.5}$$

2.2 THE GENERIC DTD

Double-talk detectors basically operate in the same manner. Thus, the general procedure for handling double-talk is described by the following:

1. A detection statistic ξ is formed using available signals, e.g. x, y, e, etc., and the estimated filter coefficients $\hat{h}$.

2. The detection statistic ξ is compared to a preset threshold *T,* and double-talk is declared if $\xi < T$.

3. Once double-talk is declared, the detection is held for a minimum period of time T_{hold}. While the detection is held, the filter adaptation is disabled.

4. If $\xi \geq T$consecutively over a time T_{hold}, the filter resumes adaptation, while the comparison of ξ to T continues until $\xi < T$ again.

The hold time T_{hold} in Step 3 and Step 4 is necessary to suppress detection dropouts due to the noisy behavior of the detection statistic. Although there are some possible variations, most of the DTD algorithms keep this basic form and only differ in how to form the detection statistic.

An "optimum" decision variable ξ for double-talk detection will behave as follows:

(i) if $v(n) = 0$ (double-talk is not present), $\xi \geq T$.

(ii) if $v(n) \neq 0$ (double-talk is present), $\xi < T$.

(iii) ξ is insensitive to echo path variations.

The threshold T must be a constant, independent of the data. Moreover, it is desirable that the decisions are made without introducing any delay (or minimize the introduced delay) in the updating of the adaptive filter. The delayed decisions will otherwise affect the AEC algorithm negatively.

2.3 A SUGGESTION TO PERFORMANCE EVALUATION OF DTDS

The role of the threshold T is essential to the performance of the double-talk detector. To select the value of T and to compare different DTDs objectively one could view the DTD as a classical binary detection problem. By doing so, it is possible to rely on established detection theory. This approach to characterize DTDs was proposed in [4, 6].

The general characteristics of a binary detection scheme are:

- *Probability of False Alarm* (P_{f}): Probability of declaring detection when a target, in our case double-talk, is not present.
- *Probability of Detection* (P_{d}): Probability of successful detection when a target is present.
- *Probability of Miss* ($P_{\text{m}} = 1 - P_{\text{d}}$): Probability of detection failure when a target is present.

A well designed DTD maximizes P_{d} while minimizing P_{f} even in a low SNR. In general, higher P_{d} is achieved at the cost of higher P_{f}. There should be a tradeoff in performance depending on the penalty or cost function of a false alarm.

One common approach to characterize different detection methods is to represent the detection characteristic P_{d} (or P_{m}) as a function of false alarm probability P_{f} under a given constraint on the SNR. This is known as a receiver operating characteristic (ROC). The P_f constraint can be interpreted as the maximum tolerable false alarm rate.

Evaluation of a DTD is carried out by estimating the performance parameters, P_d (P_m) and P_f. A principle for this technique can be found in [6]. Though in the end, one should accompany these performance measures with a joint evaluation of the DTD and the AEC. This is due to the fact that the response time of the DTD can seriously affect the performance of the AEC and this is in general not shown in the ROC curve.

3. DOUBLE-TALK DETECTION ALGORITHMS

In this section, we explain different DTD algorithms that can be useful for AEC. We start with the Geigel algorithm since it was the very first DTD proposal.

3.1 THE GEIGEL ALGORITHM

A very simple algorithm due to A. A. Geigel [8] is to declare the presence of near-end speech whenever

$$\xi^{(\mathrm{g})} = \frac{\max\{\,|x(n)|, \ldots, |x(n-L_\mathrm{g}+1)|\,\}}{|y(n)|} < T, \tag{6.6}$$

where L_g and T are suitably chosen constants. This detection scheme is based on a waveform level comparison between the microphone signal $y(n)$ and the far-end speech $x(n)$ assuming the near-end speech $v(n)$ in the microphone signal will be typically stronger than the echo $\mathbf{h}^T\mathbf{x}$. The maximum or l_∞ norm of the L_g most recent samples of $x(n)$ is taken for the comparison because of the undetermined delay in the echo path. The threshold T is to compensate for the energy level of the echo path response h, and is often set to 2 for network echo cancelers because the hybrid loss is typically about 6 dB or more. For an AEC, however, it is not clear how to set a universal threshold to work reliably in all the various situations because the loss through the acoustic echo path can vary greatly depending on many factors. For L_g, one choice is to set it the same as the adaptive filter length L since we can assume that the echo path is covered by this length.

3.2 THE CROSS-CORRELATION METHOD

In [9] the cross-correlation coefficient vector between $\mathbf{x}$ and e was proposed as a means for double-talk detection. A similar idea using the cross-correlation coefficient vector between $\mathbf{x}$ and y has proven more robust and reliable [10, 6]. This section will therefore focus on the cross-correlation coefficient vector between $\mathbf{x}$ and y which is defined as

$$
\begin{aligned}
\mathbf{c}_{xy}^{(1)} &= \frac{E\{\mathbf{x}(n)y(n)\}}{\sqrt{E\{x^2(n)\}E\{y^2(n)\}}} \\
&= \frac{\mathbf{r}_{xy}}{\sigma_x\sigma_y} \\
&= \left[\begin{array}{cccc} c_{xy,0}^{(1)} & c_{xy,1}^{(1)} & \cdots & c_{xy,L-1}^{(1)} \end{array}\right]^T,
\end{aligned} \tag{6.7}
$$

where $c_{xy,i}^{(1)}$ is the cross-correlation coefficient between $x(n-i)$ and $y(n)$.

The idea here is to compare

$$
\begin{aligned}
\xi^{(1)} &= \|\mathbf{c}_{xy}^{(1)}\|_\infty \\
&= \max_i |c_{xy,i}^{(1)}|, \; i = 0, 1, ..., L-1
\end{aligned} \tag{6.8}
$$

to a threshold level T. The decision rule will be very simple: if $\xi^{(1)} \geq T$, then double-talk is not present; if $\xi^{(1)} < T$, then double-talk is present.

(Although the l_∞ norm used in (6.7) is perhaps the most natural, other scalar metrics, e.g., l_1, l_2, could alternatively be used to assess the cross-correlation coefficient vectors. However, there is a fundamental problem here which is not linked to the type of metric used. The problem is that these cross-correlation coefficient vectors are not well normalized. Indeed, we can only say in general that $\xi^{(1)} \leq 1$. If $v(n) = 0$, that does not imply that $\xi^{(1)} = 1$ or any other known value. We do not know the value of $\xi^{(1)}$ in general. The amount of correlation will depend a great deal on the statistics of the signals and of the echo path. As a result, the best value of T will vary a lot from one experiment to another. So there is no natural threshold level associated with the variable $\xi^{(1)}$ when $v(n) = 0$.

Next section presents a decision variable that exhibits better properties than the cross-correlation algorithm. This decision variable is formed by properly normalizing the cross-correlation vector between $\mathbf{x}$ and y.

3.3 THE NORMALIZED CROSS-CORRELATION METHOD

There is a simple way to normalize the cross-correlation vector between a vector $\mathbf{x}$ and a scalar y in order to have a natural threshold level for ξ when $v(n) = 0$.

Suppose that $v(n) = 0$. In this case:

$$\sigma_y^2 = \mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}, \tag{6.9}$$

where $\mathbf{R}_{xx} = E\{\mathbf{x}(n)\mathbf{x}^T(n)\}$. Since $y(n) = \mathbf{h}^T\mathbf{x}(n)$, we have

$$\mathbf{r}_{xy} = \mathbf{R}_{xx}\mathbf{h}, \tag{6.10}$$

and (6.9) can be re-written as

$$\sigma_y^2 = \mathbf{r}_{xy}^T \mathbf{R}_{xx}^{-1} \mathbf{r}_{xy}. \tag{6.11}$$

In general for $v(n) \neq 0$ we have,

$$\sigma_y^2 = \mathbf{r}_{xy}^T \mathbf{R}_{xx}^{-1} \mathbf{r}_{xy} + \sigma_v^2. \tag{6.12}$$

If we divide (6.11) by (6.12) and compute its square root, we obtain the decision variable [5, 14]

$$\begin{aligned} \xi^{(2)} &= \sqrt{\mathbf{r}_{xy}^T (\sigma_y^2 \mathbf{R}_{xx})^{-1} \mathbf{r}_{xy}} \\ &= \|\mathbf{c}_{xy}^{(2)}\|_2, \end{aligned} \tag{6.13}$$

where

$$\mathbf{c}_{xy}^{(2)} = (\sigma_y^2 \mathbf{R}_{xx})^{-1/2} \mathbf{r}_{xy} \tag{6.14}$$

is what we will call the normalized cross-correlation vector between **x** and y.

Substituting (6.10) and (6.12) into (6.13), we show that the decision variable is:

$$\xi^{(2)} = \frac{\sqrt{\mathbf{h}^T \mathbf{R}_{xx} \mathbf{h}}}{\sqrt{\mathbf{h}^T \mathbf{R}_{xx} \mathbf{h} + \sigma_v^2}}. \tag{6.15}$$

We easily deduce from (6.15) that for $v(n) = 0$, $\xi^{(2)} = 1$ and for $v(n) \neq 0$, $\xi^{(2)} < 1$. Note also that $\xi^{(2)}$ is not sensitive to changes of the echo path when $v = 0$.

For the particular case when x is white Gaussian noise, the autocorrelation matrix is diagonal: $\mathbf{R}_{xx} = \sigma_x^2 \mathbf{I}$. Then (6.14) becomes:

$$\begin{aligned} \mathbf{c}_{xy}^{(2)} &= \frac{\mathbf{r}_{xy}}{\sigma_x \sigma_y} \\ &= \mathbf{c}_{xy}^{(1)}. \end{aligned} \tag{6.16}$$

Hence, in general what we are doing in (6.13) is equivalent to prewhitening the signal **x**, which is one of many known “generalized cross-correlation” techniques [15]. Thus, when **x** is white, no prewhitening is necessary and $\mathbf{c}_{xy}^{(2)} = \mathbf{c}_{xy}^{(1)}$. This suggests a more practical implementation, whereby matrix operations are replaced by an adaptive prewhitening filter [16].

Finally, a fast version of (6.15) can be derived by recursively updating $\mathbf{R}_{xx}^{-1} \mathbf{r}_{xy}$ using the Kalman gain $\mathbf{R}_{xx}^{-1} \mathbf{x}$ [3]. Estimated quantities of the cross-correlation and the near-end signal power have to be introduced for the derivation of a fast version. Equation (6.15) can be rewritten as

$$\xi^2(n) = \frac{\mathbf{r}^T(n) \mathbf{R}^{-1}(n) \mathbf{r}(n)}{\sigma_y^2(n)} = \frac{\eta^2(n)}{\sigma_y^2(n)}, \tag{6.17}$$

Table 6.1 The fast version of the NCC double-talk. Note that we need to distinguish between the "background" echo path estimate calculated in the DTD, denoted $\hat{\mathbf{h}}_{\mathrm{b}}(n)$, and the estimate calculated in the echo canceler, $\hat{\mathbf{h}}(n)$. The a postriori Kalman gain is $\mathbf{k}'(n) =)\mathbf{R}^{-1}(n)\mathbf{x}(n)$.

Double-talk detector

$$\begin{aligned}
\sigma_y^2(n) &= \lambda\sigma_y^2(n-1) + y^2(n) \\
e_{\mathrm{b}}(n) &= y(n) - \mathbf{h}_{\mathrm{b}}^T(n-1)\mathbf{x}(n) \\
\eta^2(n) &= \lambda\eta^2(n-1) + y^2(n) - \varphi(n)e_{\mathrm{b}}^2(n) \\
\eta(n)/\sigma_y(n) &< T, \Rightarrow \text{ double-talk, } \mu = 0 \\
\eta(n)/\sigma_y(n) &\geq T, \Rightarrow \text{ no double-talk, } \mu = 1 \\
\mathbf{h}_{\mathrm{b}}(n) &= \mathbf{h}_{\mathrm{b}}(n-1) + \mathbf{k}'(n)\frac{e_{\mathrm{b}}(n)}{\varphi(n)}
\end{aligned}$$

where we squared the statistic for simplicity. The correlation variables are estimated recursively as,

$$\begin{aligned}
\mathbf{r}(n) &= \lambda\mathbf{r}(n-1) + \mathbf{x}(n)y(n), && (6.18) \\
\mathbf{R}(n) &= \lambda\mathbf{R}(n-1) + \mathbf{x}(n)\mathbf{x}^T(n), && (6.19) \\
\sigma_y^2(n) &= \lambda\sigma_y^2(n-1) + y^2(n), && (6.20) \\
\alpha(n) &= \lambda + \mathbf{x}^T(n)\mathbf{R}^{-1}(n-1)\mathbf{x}(n), && (6.21)
\end{aligned}$$

where $\lambda < 1$ is a forgetting factor. The statistic $\eta^2(n)$ can be shown to be updated as

$$\eta^2(n) = \lambda\eta^2(n-1) + y^2(n) - \varphi(n)e^2(n)\ , \qquad (6.22)$$

where the likelihood variable $\varphi(n) = \lambda/\alpha(n)$ and $e(n)$ is the residual error, $e(n) = y(n) - \hat{y}(n)$. Hence, the quantities required to form the test statistic of the fast version of the NCC DTD are given by the simple first-order recursions in (6.20) and (6.22).

Table 6.1 gives the calculations for the fast NCC DTD, where it is assumed that the Kalman gain has been calculated "for free" by the FRLS algorithm [17].

3.4 THE COHERENCE METHOD

Instead of using the cross-correlation vector, a detection statistic can be formed by using the squared magnitude coherence. A DTD based on coherence was proposed in [4]. The idea is to estimate the coherence between $\mathbf{x}(n)$ and $\mathbf{y}(n)$. The coherence is close to one when there is no double-talk and it is close

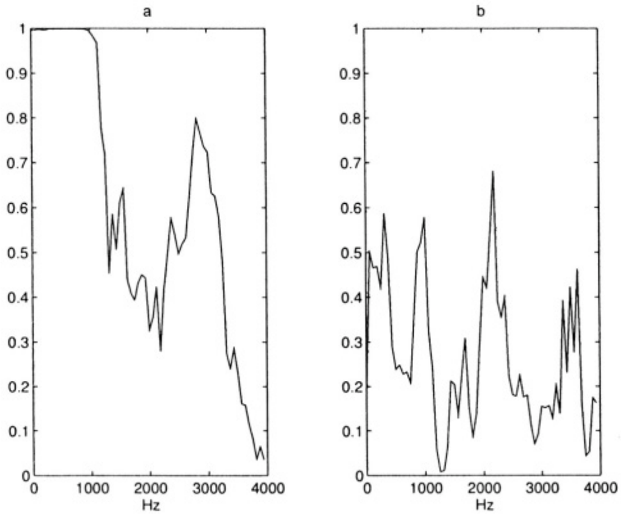


Figure 6.2 Estimated coherence using the multiple window method. (a) Far-end talker is active only. (b) Double-talk situation where the far- and near-end signals powers are equal. Echo path attenuation is 6 dB and the ambient noise power is 25 dB lower than the far-end speech.

to zero in a double-talk situation. Figure 6.2 shows an example of estimated coherence between loudspeaker and microphone signals in the presence and absence of double-talk. The squared coherence is defined as,

$$\gamma_{xy}^2(k) = \frac{|S_{xy}(k)|^2}{S_{xx}(k)S_{yy}(k)}, \tag{6.23}$$

where $S_{..}(k)$ is the DFT based cross-power spectrum and k is the DFT frequency index. As decision parameter, an average over a few frequencies is used as detection statistics,

$$\xi^{(3)} = \frac{1}{I}\sum_{i=0}^{I-1}\gamma_{xy}^2(k_i), \tag{6.24}$$

where I is the number of intervals used. Typical choices of these parameters are $I = 3$ and k_0, k_1, k_2 are the intervals chosen such that their center correspond to approximately 300, 1200, and 1800 Hz respectively. This gives in practice a significantly better performance than averaging over the whole frequency range since there is a poorer speech-to-noise ratio in the upper frequencies (the average speech spectrum declines with about 6 dB/octave above 2 kHz).

Estimation of the spectra in (6.23) can be made by using the multiple window technique [18], where

$$\hat{S}_{xx}(k) = \frac{1}{P}\sum_{p=0}^{P-1} |X_p(k)|^2, \tag{6.25}$$

$$\hat{S}_{xy}(k) = \frac{1}{P}\sum_{p=0}^{P-1} X_p(k)Y_p^*(k), \tag{6.26}$$

where $X_p(k)$ is the $p : th$ eigenspectrum

$$X_p(k) = \sum_{n=0}^{L_c-1} x(L_c - 1 - n)\phi_p(n)e^{-j2\pi\frac{k}{L_c}n} \tag{6.27}$$

and $Y_p(k)$ is analogously defined. The window $\phi_p(n)$ is the $p : th$ discrete spheroidal wave function [19]. L_c is the block length of the DFT. The multiple window method has advantages such as easy tradeoff between bias and variance. Another possibility is to use the Welch spectrum estimation method [20].

Since this DTD is based on block processing of the signals, there is a tradeoff between calculation complexity and time between decisions. It is desirable to keep the time between decisions as short as possible in order to have as low detection failures as possible (both false alarm and detection miss).

3.5 THE NORMALIZED CROSS-CORRELATION MATRIX

Obviously, the cross-correlation and coherence methods are related in some sense. This link can be established by extending the definition of the cross-correlation method to incorporate correlation between two vectors **x** and **y** instead of only the scalar $y(n)$ [5]. Define the normalized cross-correlation matrix $\mathbf{C}_{xy}$ between two vectors **x** and **y** as follows

$$\mathbf{C}_{xy} = \mathbf{R}_{xx}^{-1/2}\mathbf{R}_{xy}\mathbf{R}_{yy}^{-1/2}, \tag{6.28}$$

where

$$\mathbf{y}(n) = \left[\begin{array}{cccc} y(n) & y(n-1) & \cdots & y(n-N+1) \end{array}\right]^T$$

is a vector of size N. There are two interesting cases:

(i) $N = 1$, $\mathbf{C}_{xy} = \mathbf{c}_{xy}^{(2)}$ (normalized cross-correlation vector between **x** and y).

(ii) $N = L = 1$, $\mathbf{C}_{xy} = c_{xy,0}^{(1)}$ (cross-correlation coefficient between x and y).

By extension to (6.13), we then form the detection statistic

$$\xi^{(4)} = \frac{1}{\sqrt{N}}\|\mathbf{C}_{xy}\|_{\mathrm{F}} = \frac{1}{\sqrt{N}}\sqrt{\mathrm{tr}(\mathbf{C}_{xy}^T\mathbf{C}_{xy})}, \tag{6.29}$$

where the subscript "F" denotes the Frobenius norm. We note that for case (i), $\xi^{(4)} = \xi^{(2)}$ as before. Again, we can interpret this formulation as a "generalized cross-correlation", where now both **x** and **y** are prewhitened, which is also known as the "smoothed coherence transform" (SCOT) [15].

The link between the normalized cross-correlation matrix and the coherence can now be established as follows: Suppose that $N = L \to \infty$. In this case, a Toeplitz matrix is asymptotically equivalent to a circulant matrix if its elements are absolutely summable [21], which is the case for the intended application. Hence we can decompose $\mathbf{R}_{ab}$ as

$$\mathbf{R}_{ab} = \mathbf{F}^{-1}\mathbf{S}_{ab}\mathbf{F}, \tag{6.30}$$

where **F** is the discrete Fourier transform (DFT) matrix and

$$\mathbf{S}_{ab} = \mathrm{diag}\{S_{ab}(0), S_{ab}(1), \cdots, S_{ab}(L-1)\} \tag{6.31}$$

is a diagonal matrix formed by the first column of $\mathbf{FR}_{ab}$, and

$$\begin{aligned} S_{ab}(k) &= \sum_{m=-\infty}^{+\infty} E\{a(n)b(n-m)\}\mathrm{e}^{-i2\pi km/L} \\ &= \sum_{m=-\infty}^{+\infty} R_{ab}(m)\mathrm{e}^{-i2\pi km/L} \end{aligned} \tag{6.32}$$

is the DFT cross-power spectrum. Now:

$$\begin{aligned} \mathrm{tr}(\mathbf{C}_{xy}^T\mathbf{C}_{xy}) &= \mathrm{tr}(\mathbf{R}_{yy}^{-1/2}\mathbf{R}_{yx}\mathbf{R}_{xx}^{-1}\mathbf{R}_{xy}\mathbf{R}_{yy}^{-1/2}) \\ &= \mathrm{tr}(\mathbf{R}_{yx}\mathbf{R}_{xx}^{-1}\mathbf{R}_{xy}\mathbf{R}_{yy}^{-1}) \end{aligned} \tag{6.33}$$

since tr(**AB**) = tr(**BA**). Using (6.30), we easily find that

$$\begin{aligned} \mathrm{tr}(\mathbf{C}_{xy}^T\mathbf{C}_{xy}) &= \mathrm{tr}(\mathbf{S}_{yx}\mathbf{S}_{xx}^{-1}\mathbf{S}_{xy}\mathbf{S}_{yy}^{-1}) \\ &= \sum_{k=0}^{L-1} |\gamma_{xy}(k)|^2, \end{aligned} \tag{6.34}$$

where

$$\gamma_{xy}(k) = \frac{S_{xy}(k)}{\sqrt{S_{xx}(k)S_{yy}(k)}} \tag{6.35}$$

is the discrete coherence function. Thus, asymptotically we have

$$\begin{aligned}\xi^{(4)} &\approx \sqrt{\frac{1}{L}\sum_{k=0}^{L-1}|\gamma_{xy}(k)|^2} \\ &= \sqrt{\frac{1}{L}\sum_{k=0}^{L-1}\frac{|H(k)|^2}{|H(k)|^2+\kappa(k)}},\end{aligned} \tag{6.36}$$

where $H(k)$ is the transfer function of h and

$$\kappa(k) = \frac{S_{vv}(k)}{S_{xx}(k)} \geq 0 \tag{6.37}$$

is the near-end talker to far-end talker spectral ratio at frequency k. Except for an unrestricted frequency range, this form is identical to the coherence-based double-talk detector presented in Section 3.4. We find that this idea is very appropriate since when $v(n) = 0$, the two signals x and y are completely coherent and then $|\gamma_{xy}(k)| = 1, \forall k$, and $\xi^{(4)} \approx 1$; when $v \neq 0$, $|\gamma_{xy}(k)| < 1, \forall k$, and $\xi^{(4)} < 1$.

3.6 THE TWO-PATH MODEL

An interesting approach to double-talk handling was proposed in [13]. This method was introduced for network echo cancellation. However, it has proven far more useful for the AEC application. In this method, two filters model the echo path, one background filter which is adaptive as in a conventional AEC solution and one foreground filter which is not adaptive. The foreground filter cancels the echo. Whenever the background filter performs better than the foreground, its coefficients are copied to the foreground. Coefficients are copied only when a set of conditions are met, which should be compared to the single statistic decision declaring "no double-talk" in a traditional DTD presented in the previous sections.

The basic set of conditions found in [13] are given by (6.38)-(6.40). Copying is performed, equivalent to no double-talk present, if any of (6.38)-(6.40) is fulfilled,

$$\xi^{(y)} = \frac{P(e_{\mathrm{b}})}{P(y)} < T_y \tag{6.38}$$

$$\xi^{(e)} = \frac{P(e_{\mathrm{b}})}{P(e_{\mathrm{f}})} < T_{\mathrm{e}} \tag{6.39}$$

$$\xi^{(x)} = \frac{P(y)}{P(x)} < 1 \tag{6.40}$$

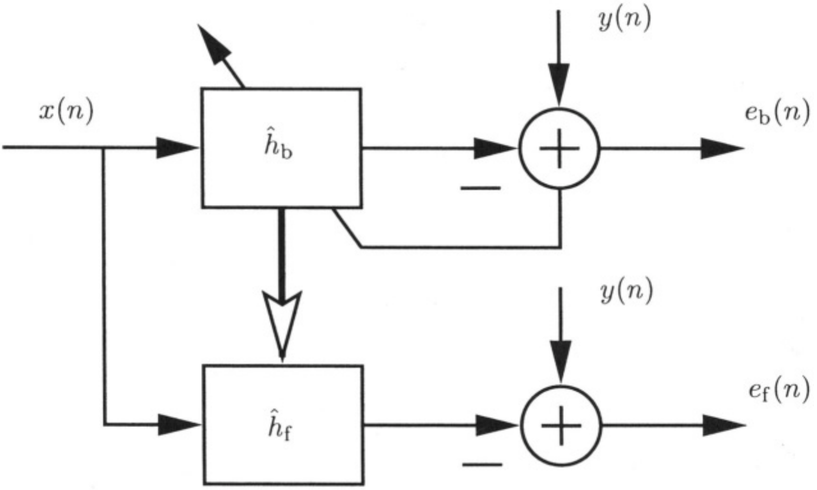


Figure 6.3 Two-path adaptive filtering. The adaptive filter estimates the impulse response $\hat{h}_b$. If, according to some criteria, $\hat{h}_b$ is determined to be a better estimate than the earlier estimate $\hat{h}_f$, the coefficients in $\hat{h}_b$ are copied to $\hat{h}_f$. The latter is then used to calculate the residual echo signal $e_f(n)$.

where $P(a)$ is the short time smoothed absolute magnitude of a signal $a(n)$,

$$P(a(n)) = \sum_{l=0}^{L_{TP}-1} |a(n-l)|. \tag{6.41}$$

A hangover time T_{hold} is also imposed when (6.40) is fulfilled. The last condition (6.40) is basically the same as in the Geigel DTD with a unity threshold, i.e., the echo path is assumed not to attenuate the far-end speech. If all three conditions are satisfied over *D* consecutive decisions, copying of background coefficients is resumed.

Condition (6.38) ensures the background adaptive filter is canceling echo, while condition (6.39) ensures the background filter is outperforming the foreground filter. The above decision logic is effective for certain applications, but is not without shortcomings. First, conditions (6.38) and (6.39) are not always sufficient to prevent coefficient transfer in the presence of double-talk and/or high background noise. For speech or any other non-spectrally diverse excitation, the inequalities in (6.38) and (6.39) can be satisfied in the short term (over duration *D* for example) even though the actual misalignment error of the background coefficients is worse than that of the foreground coefficients. Second, (6.38) and (6.39) employ thresholds that limit the responsiveness of the logic to changes in the performance of the background canceler and to changes in the physical echo path. Condition (6.38) requires the background canceler to achieve a certain degree (18 dB in [13]) of cancellation before the foreground can be updated. In the presence of an echo path change, for example, (6.38) can prolong the presence of annoying echo. The threshold in (6.39) ensures that the foreground is updated only in steps, in effect quantizing the convergence trajectory of the echo canceler. Last, condition (6.40) ensures no update is performed unless $|y(n)| < |x(n)|$. But, this property can be used to inhibit adaptation only in cases for which the physical echo path introduces signal loss ($\mathbf{h}^T\mathbf{h} < 1$). If the echo path introduces gain ($\mathbf{h}^T\mathbf{h} > 1$), condition (6.40) prevents adaptation even in the absence of near-side speech and noise. For this reason, these rules cannot in general be used in echo-canceling speakerphones, where $\mathbf{h}^T\mathbf{h} > 1$.

3.6.1 A Threshold-Free Decision Logic. In addition to the beneficial aspects of the original two-path logic, a two-path canceler decision logic should possess the following characteristics:

- Faster initial convergence and reconvergence following echo path changes.
- Applicability to echo paths having signal gain ($\mathbf{h}^T\mathbf{h} > 1$).
- Reduced dependence upon user-selected constants, such as thresholds and timers.

A logic that exhibit these properties is proposed and described in detail in [22]. This decision logic differs from that of prior works in that it does not use decision thresholds (constants). A smoothing parameter is the only constant that has to be chosen. Moreover, the logic applies to both lossy and gain-incurring echo paths, and possesses favorable convergence properties for many scenarios encountered in practice. Hence, the great advantage with this algorithm is that it is not sensitive to echo path changes since the background filter is allowed to track changes freely and as soon as it performs better than the foreground it is copied over.

3.7 DTD COMBINATIONS WITH ROBUST STATISTICS

All practical double-talk detectors have a probability of miss, i.e. $P_{\mathrm{m}} \neq 0$. Requiring the probability of miss to be smaller will undoubtedly increase the probability of false alarms hence slowing down the convergence rate. As a consequence, no matter what DTD is used, undetected near-end speech will perturb the adaptive algorithm from time to time. Figure 6.4 shows the remaining undetected near-end speech (double-talk) after double-talk detection with a Geigel detector with $T = 2$. The impact of this perturbation is governed by the echo to near-end speech ratio as described in Section 1.

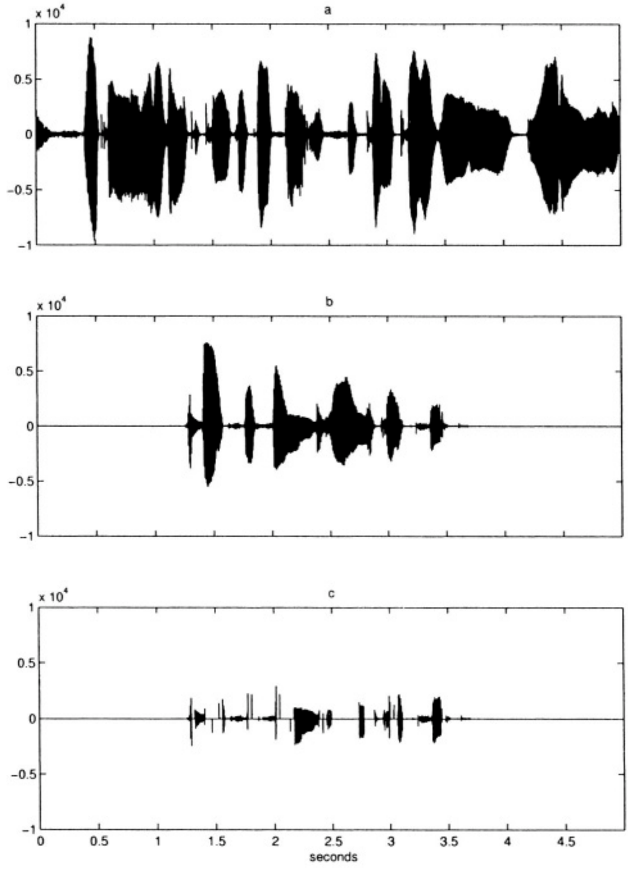


Figure 6.4 (a) Far-end speech. (b) Near-end speech, i.e. double-talk. (c) Near-end speech gated with the decision of the DTD. These are the disturbances that actually enters the adaptive algorithm. Average far- to near-end ratio: 6 dB (1.125-3.625 s).

In practice, what has been done in the past is, first the DTD is designed to be "as good as" one can afford and then, the adaptive algorithm is slowed down so that it copes with the detection errors made by the DTD. This is natural to do since if the adaptive algorithm is very fast, it can react faster to situation changes (e.g. double-talk) than the DTD and thus can diverge. However, this approach severely penalizes the convergence rate of the AEC when the situation is good, i.e. far-end but no near-end talk is present.

In the light of these facts, it may be fruitful to look at adaptive algorithms that can handle at least a small amount of double-talk without diverging. This approach has been studied and proven very successful in the network echo canceler case [23], where the combination of outlier resistant adaptive algorithms and a Geigel DTD were studied. For the acoustic case, one could use any appropriate DTD and combine it with a robust adaptive algorithm,

The approach can be exemplified by a robust version of the NLMS algorithm:

$$\begin{aligned}\hat{\mathbf{h}}(n) &= \hat{\mathbf{h}}(n-1) \\ &+ \frac{\mu_0 \mathbf{x}(n)}{\mathbf{x}^T(n)\mathbf{x}(n)+\delta}\psi\left[\frac{|e(n)|}{s(n-1)}\right] \operatorname{sign}\left[e(n)\right] s(n-1). \quad (6.42)\end{aligned}$$

As for any AEC/DTD, adaptation is inhibited by setting the step-size parameter to zero when double-talk is detected. The scaled non-linearity $\psi(\cdot)$ in (6.41) can be chosen to be the limiter [24],

$$\psi\left[\frac{|e(n)|}{s(n)}\right] = \min\left[\frac{|e(n)|}{s(n)}, k_0\right], \quad (6.43)$$

where $s(n)$ is an adaptive scale factor. Making the scale factor adaptive and supervised by the DTD is the key to the success of this approach. The scale factor should reflect the background noise level at the near-end, be robust to short burst disturbances (double-talk) and track long term changes of the residual error (echo path changes). To fulfill these requirements one can choose the scale factor estimate as

$$s(n) = \lambda s(n-1) + \frac{1-\lambda}{\beta} s(n-1)\psi\left[\frac{|e(n)|}{s(n-1)}\right], \quad (6.44)$$

where $s_{-1} = \sigma_x$. Adaptation of $s(n)$ is performed as long as the DTD has not detected double-talk. Justification and details of the above derivations can be found in [23].

4. COMPARISON OF DTDS BY MEANS OF THE ROC

In this section, we present receiver operating characteristics of three DTDs, namely, the Geigel detector, the cross-correlation detector, and the normalized cross-correlation detector. As reference, we also show the region of operation for the "threshold free" two-path logic described in Section 3.6.

Estimates of P_{m} and P_{f} were obtained according to the procedure in [6] and we present the ROCs estimated from speech as well as stationary synthetic data. The speech data we use contain sentences from three male and two female talkers. Furthermore, all sentences/synthetic data are also normalized to have the same average power level.

Simulation details are as follows:

Echo path. The echo path used is a measured acoustic response between the left loudspeaker and a standard cardioid microphone positioned on top of a workstation. The original impulse response has a length of 256 ms, consisting of 4096 coefficients at a 16 kHz sampling rate. However, it was subsequently decimated to an 8 kHz sampling rate, resulting in 2048 coefficients. It is also normalized so that $\sigma_{y_e}^2 = \sigma_x^2$ for the actual speech and synthetic data.

Probability of false alarm. When estimating the probability of false alarm we use sentences from five talkers as far-end speech. These consist of speech sequences of 21.8 seconds at an 8 kHz sampling rate. The echo-to-ambient-noise ratio, $\mathrm{ENR} = \sigma_{y_e}^2/\sigma_w^2$, is set to 1000 (30 dB).

Probability of miss. The probability of a miss is estimated using 5 seconds of far-end speech from one male talker. As near-end speech 8 sentences are used, each about 2 seconds long. In this case, we investigate the performance when the average echo-to-background ratio $\mathrm{EBR} = \sigma_{y_e}^2/(\sigma_v^2 + \sigma_w^2) = 1$ (0 dB), since it is natural to assume equally strong talkers.

The simulation conditions for the case with synthetic data are equivalent to those of the speech case. The synthetic sequences [far-end, near-end (double-talk), and ambient noise] are all white Gaussian distributed and mutually independent. The synthetic data enables us to assess the influence of in-stationarity of speech (e.g. the instantaneously varying EBR/ENR).

A hold time (T_{hold}) of 30 ms (240 samples) is used in all three detectors. The thresholds of the detectors are chosen such that their probability of false alarm is in the range of 0 to about 1. For these thresholds, we then estimate the corresponding probability of detection (see [6] for details). Since the two-path method is threshold free, we instead vary the smoothing parameter over a range of practical values and estimate the resulting probabilities. These probabilities are presented together with the ROCs of the other detectors. The smoothing parameter (time constant) is varied from 0.1 to 0.5 s in steps of 0.1 s.

Results from these simulations are shown in Fig. 6.5. These results are consistent with those reported in [6] and, by the use of the ROC, it is possible to set thresholds such that the DTDs are compared fairly. In general, P_{m} increases and P_{f} decreases when we compare the ROC curves estimated using speech versus the ROC curves estimated using synthetic signals. We also find that the ROC of the Geigel detector is more sensitive to this signal condition change than the ROCs of the other detectors. It is clear from these results that the normalized cross-correlation detector has superior performance compared to the two others. Also in this figure, the operation region for the threshold free two-path implementation is shown. The two-path method takes into account whether or not it is beneficial to update the adaptive algorithm for the specific data set. Hence, the estimated probability of false alarm increases and this should be taken into account when interpreting the results. The lowest probability of miss is attained when the smoothing parameter (time constant) is 0.5 s.

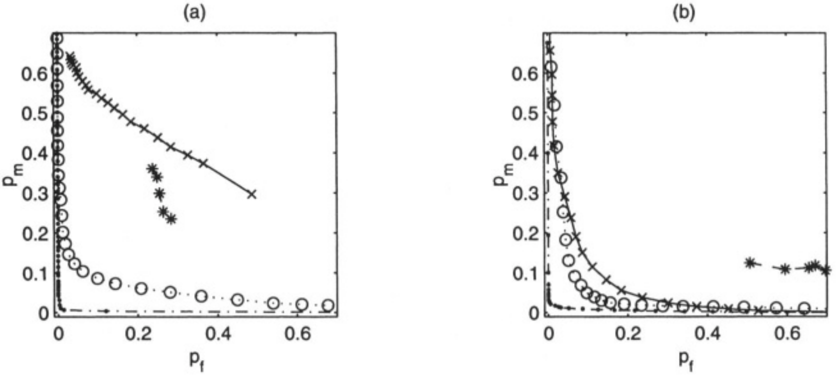


Figure 6.5 **Receiver operating characteristic (ROC) (a) based on speech, (b) based on synthetic data. Double-talk detectors: Geigel (×), cross-correlation (o), and normalized cross-correlation (·). Region of operation for the threshold free two-path logic is indicated by (∗). EBR = 0 dB and ENR = 30 dB.**

5. DISCUSSION

In this chapter, we have presented double-talk detection algorithms suitable for acoustic echo cancellation. Because of the often unknown attenuation and the continuously time-varying nature of acoustic echo paths, devising an appropriate DTD is more challenging than in the network echo canceler case. There are basically two types of double-talk detectors. First, those which form their test statistics from estimated level or power of far-end, near-end including echo, or residual echo signals. Secondly, detectors that make their decisions from cross-correlation or coherence estimates of the same involved signals. In this group we also find detectors utilizing the estimate of the echo path since these estimates are derived through cross-correlation as well. Double-talk detectors based on cross-correlation techniques exhibit desirable properties needed for the acoustic case. Mainly, they have very low sensitivity to the attenuation of the echo path. However, a problem that has to be considered and designed for, is the longer response time which may result. This is due to the fact that good (low variance) test statistics need to be based on a large amount of data.

The ideal double-talk detector should be insensitive to echo path variations, have equal performance whether the echo canceler has converged or not, have quick response time and be sensitive to low near-end speech levels. Moreover, the DTD must not slow down convergence rate of the AEC which can result either from erroneous decisions or introduction of delays. Some of these properties can be characterized by probability of detection and false alarm.

Notes

1. Originally from [1]. This quotation was borrowed from [2].

References

[1] S. B. Weinstein, “Echo cancellation in the telephone network,” *IEEE Commun. Soc. Mag.,* pp. 9-15, 1977.

[2] C. R. Johnson, “On the interaction of adaptive filtering, identification, and control,” *IEEE Signal Proc. Mag.,* vol. 12, pp. 22-37, Mar. 1995.

[3] S. Haykin, *Adaptive Filter Theory.* New Jersey: Prentice-Hall, Inc, 2002.

[4] T. Gänsler, M. Hansson, C.-J. Ivarsson, and G. Salomonsson, “A double-talk detector based on coherence,” *IEEE Trans. Commun.,* vol. 44, pp. 1421-1427, Nov. 1996.

[5] J. Benesty, D. R. Morgan, and J. H. Cho, “An new class of doubletalk detectors based on cross-correlation,” *IEEE Trans. Speech Audio Processing.* vol. 8, pp 168-172, Mar. 2000.

[6] J. H. Cho, D. R. Morgan, and J. Benesty, “An objective technique for evaluating doubletalk detectors in acoustic cancelers,” *IEEE Trans. Speech Audio Processing,* vol. 7, pp. 718-724, Nov. 1999.

[7] M. M. Sondhi, “An adaptive echo canceler,” *Bell Syst. Techn. J.,* vol. XLVI, pp. 497-510, Mar. 1967.

[8] D. L. Duttweiler, “A twelve-channel digital echo canceler,” *IEEE Trans. Commun.,* vol. 26, pp. 647-653, May 1978.

[9] H. Ye and B. X. Wu, “A new double-talk detection algorithm based on the orthogonality theorem,” *IEEE Trans. Commun.,* vol. 39, pp. 1542-1545, Nov. 1991.

[10] R. D. Wesel, “Cross-correlation vectors and double-talk control for echo cancellation,” Unpublished work, 1994.

[11] J. Prado and E. Moulines, “Frequency-domain adaptive filtering with applications to acoustic echo cancellation,” *Ann. Télécomun.,* vol. 49, pp. 414-428, 1994.

[12] S. M. Kuo and Z. Pan, “An acoustic echo canceller adaptable during double-talk periods using two microphones,” *Acoustics Letters,* vol. 15, pp. 175-179, 1992.

[13] K. Ochiai, T. Araseki, and T. Ogihara, “Echo canceler with two echo path models,” *IEEE Trans. Commun.,* vol. COM-25, pp. 589-595, June 1977.

[14] J. Benesty, D. R. Morgan, and J. H. Cho, “A family of doubletalk detectors based on cross-correlation,” in *Proc. IWAENC,* Sept. 1999, pp. 108-111.

[15] C. H. Knapp and C. G. Carter, “The generalised correlation method for estimation of time delay,” *IEEE Trans. Acoust., Speech and Signal Processing,* vol. 24, pp. 320-327, Aug. 1976.

[16] J. R. Zeidler, “Performance analysis of LMS adaptive prediction filters,” *Proc. of the IEEE,* vol. 78, pp. 1781-1806, Dec. 1990.

[17] J. Benesty et. al. *Advances in Network and Acoustic Echo Cancellation.* Springer-Verlag, Berlin, 2001.

[18] D. J. Thomson, “Spectrum estimation and harmonic analysis,” *Proc. of the IEEE,* vol. 70, pp. 1055-1096, Sept. 1982.

[19] D. Slepian, “Prolate spheroidal wave funcions, Fourier analysis, and uncertainty-V,” *Bell Syst. Tech. J.,* vol. 40, pp. 1371-1429, 1978.

[20] P. D. Welch, "The use of fast Fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoustics,* vol. AU-15, pp. 70-73, June 1967.

[21] R. M. Gray, "On the asymptotic eigenvalue distribution of Toeplitz matrices," *IEEE Trans. Inform. Theory,* vol. IT-18, pp. 725-730, Nov. 1972.

[22] E. J. Diethorn, "Improved decision logic for two-path echo cancelers," in *Proc. IWAENC,* 2001.

[23] T. Gänsler, S. L. Gay, M. M. Sondhi, and J. Benesty, "Double-talk robust fast converging algorithms for network echo cancellation," *IEEE Trans. Speech Audio Processing.* vol. 8, pp. 656-663, Nov. 2000.

[24] P. J. Huber, *Robust Statistics.* pages 68-71, 135-138. New York: Wiley, 1981.

This page intentionally left blank

Chapter 7

THE WINEC: A REAL-TIME HANDS-FREE STEREO COMMUNICATION SYSTEM

Tomas Gänsler
Agere Systems
gaensler@agere.com

Volker Fischer
Darmstadt University of Technology, Department of Communication Technology
v.fischer@nt.tu-darmstadt.de

Eric J. Diethorn
Avaya Labs, Avaya
ejd@avaya.com

Jacob Benesty
Université du Québec, INRS-EMT
benesty@inrs-emt.uquebec.ca

Abstract A software application has been designed that runs a stereophonic acoustic echo canceler natively under Windows operating systems on personal computers: *the WinEC.* This is a major achievement since echo cancelers require that the sound card's input and output signals are time-synchronous. Synchronizing the audio streams is a great challenge in such an "asynchronous" environment as the operating system of a PC. Furthermore, stereophonic echo cancellation is significantly more complicated to handle than the monophonic case because of computational complexity, nonuniqueness of solution, and convergence problems. In this chapter we present the system design and the core algorithms we use. This system has been evaluated in point-to-point as well as multi-point communication scenarios. We regularly use the software for teleconferencing in wideband stereo audio over commercial IP networks.

Keywords: Real-Time Implementation, Hands-Free, Stereophonic, Acoustic Echo Canceler, VoIP

1. INTRODUCTION

Real-time echo cancellation requires a significant amount of computational resources. From a computational point of view, real-time implementation has usually been realized using custom-designed very large scale integration (VLSI) circuits or digital signal processors (DSPs) [1]. These processors are specifically designed for signal processing tasks. They provide parallel processing of operations and optimized pipeline structures. However, since the computational power of personal computers (PCs) has increased tremendously in the last few years, it is possible to perform very demanding real-time signal processing in this environment as well. Moreover, the PC environment permits the use of high-level programming languages, like C++, without the restrictions commonly imposed by DSPs, such as fixed-point arithmetic. The resulting source code can be easily used for implementing new algorithms and testing them in real-time without the need to port to special hardware. Furthermore, modern PC processors have SIMD (single instruction, multiple data) processing capabilities which can be used to speed optimize the program.

The objective of this chapter is to present a flexible echo-cancelling speakerphone algorithm that runs natively under the operating system (OS) on a PC. The additional hardware needed to support hands-free communication on a PC is a full-duplex capable sound card and a network adaptor, like a modem or an ethernet card. Depending on the desired operation mode, a mono or stereo microphone and loudspeakers are needed. For all, off-the-shelf hardware can be used.

This work was done when the authors were with Bell Labs, Lucent Technologies in the year of 2000. The system has previously been presented in [2, 3, 4]. Many of the original underlying research results can also be found in [5]. The echo canceler implementation provides the capability of communicating hands-free in single-channel mode (receive one and transmit one audio-stream), synthetic-stereo mode (receive two and transmit one stream), or full stereo mode (receive two and transmit two audio-streams). In the full stereo case, natural stereo is transmitted to the receiving side. In the synthetic case, synthesized stereo [6] or 3D-audio [7] is generated from the mono audio stream at an intermediate conference server. The bandwidth of the audio is 8 kHz. To accommodate different acoustic environments, the echo canceler can span acoustic paths of lengths 32, 64, 128, or 256 ms.

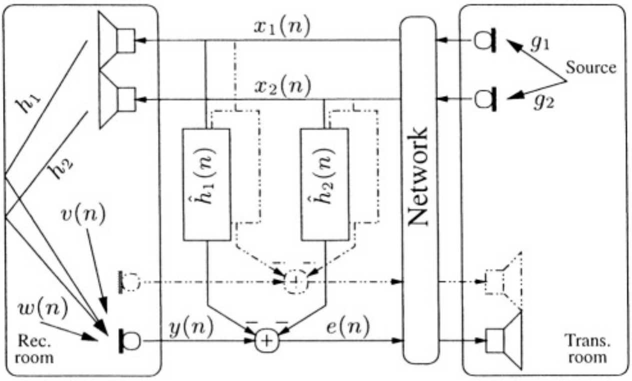


Figure 7.1 Block diagram of a generic two-channel acoustic echo canceler.

1.1 SIGNAL MODEL

A block diagram of a two-channel, point-to-point speech communication link with one[1] echo canceler is shown in Figure 7.1. We denote the signals picked up by the microphones in the transmission room by $x_1(n)$, $x_2(n)$, and the return signal picked up by one of the microphones in the receiving room by $y(n)$. The receiving room signal is in general composed of echo, ambient noise $w(n)$, and possibly receiving room speech $v(n)$. Hence, we have the receiving room signal model: $y(n) = y_e(n) + v(n) + w(n)$, where $y_e(n) = \sum_{p=1}^{2} h_p * x_p(n)$ is the echo, $*$ denotes convolution, and h_p, $p = 1, 2$, are the receiving room echo paths.

2. SYSTEM DESCRIPTION

Figure 7.2 shows a block diagram of the entire software architecture for the single-channel (mono) case running on Microsoft Windows OS. This system primarily consists of three components: the audio module, the echo cancellation module, and the network module. An overview of these modules follows.

2.1 THE AUDIO MODULE

The audio module is an interface between our software and the Windows DirectX interface [8]. DirectX provides a general interface between the Windows OS and different sound card drivers. The Windows DirectX interface is relatively well defined and stable. However, a significant concern is the so-called device driver between this interface and the actual sound card hardware. This driver is designed by the manufacturer of the sound card hardware and it is difficult to predict how it interacts with the Windows OS.

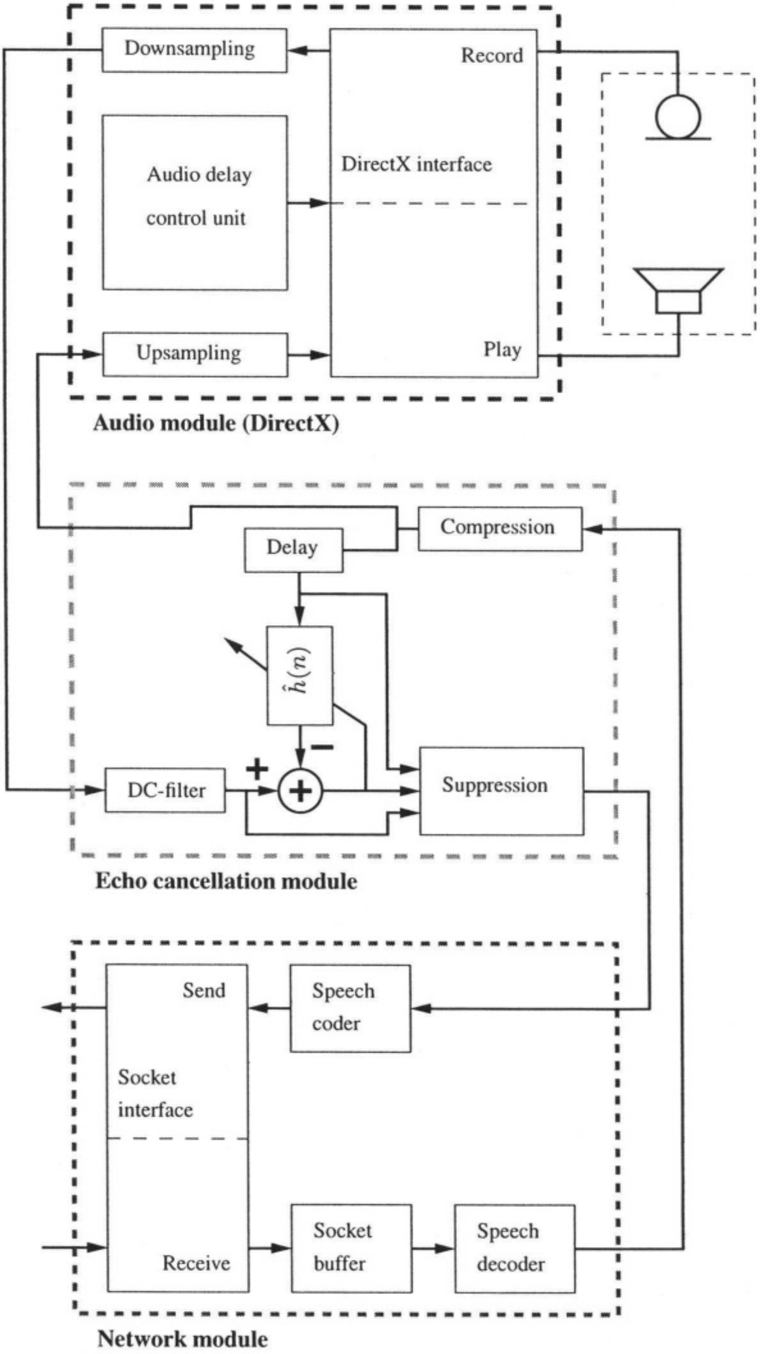


Figure 7.2 System block diagram for the single-channel case.

The key problem encountered when implementing an echo canceler on a PC is loss of synchronization of the audio streams. This causes instantaneous delay variation in the actual echo path, and the canceler cannot track these changes fast enough to achieve proper cancellation. There are two primary problems that must be solved to achieve stable echo cancellation performance. The consequences of, and solutions to, these problems are shown in the following.

2.1.1 Stream synchronization problem caused by OS sample-rate conversion. On a multitasking OS, more than one application can generate sound at a time. Thus, the OS must mix the different audio streams together. Since the streams can only be mixed if they have a common sample rate, it is necessary to apply sample-rate conversion. To avoid loss of sound quality, the common sample rate is chosen to be the highest sample rate of the different sound streams.

The problem for the sound card driver software, which realizes the sample-rate conversion, is that it is usually restricted to the use of certain block sizes. To implement an exact conversion between two sample rates, it is often necessary to change the input or output block sizes with time, e.g., when converting from 44.1 kHz to our sample rate of 16 kHz is necessary. To simplify the implementation, some manufacturers tend to omit or insert samples to keep the block sizes constant. In this case synchronization between input and output streams suffers from a constant phase drift, or jitter, which results in poor echo cancellation.

The best way to resolve this problem is to take complete control over sample-rate conversion by performing it at the application layer. To make sure the driver does not modify the signal, the application up-samples to the highest sample rate offered by the sound card. As such, no other sample rate can be chosen by the OS without incurring loss of audio quality, and the sound card does not need to apply sample-rate conversion to play and record the streams.

2.1.2 Synchronization failure following temporary bursts of CPU utilization. The general flowchart of a block-oriented, full-duplex audio application consists of three boxes: a sound card read box, which queries the audio data from the sound interface; a signal processing box where echo cancellation is done; and a third box that writes the processed data to the sound interface. The first box usually holds the data flow until a new input data block is captured by the sound card. To be sure that the application runs stable even at nearly 100% CPU usage, the minimum audio latency must not be shorter than the duration of one data block. Therefore, the third box writes the processed data exactly one block further than the play position at the time a new block is read by the sound-in box. Unfortunately, if the operating system reaches the CPU limit, the difference between read and write position in the buffers changes abruptly and stays steady at a new value. This phenomenon occurs on many sound cards. To combat this situation, a precise measurement of the current read and write position must be available to correct the synchronization failure. Unfortunately, the buffer read and write positions reported by the sound interface are inaccurate. In this implementation, the positions are queried often during regular operation and averaged to reduce the variance of the measurement when a failure happens.

2.2 THE NETWORK MODULE

The network module controls the data transfer between the two connected clients. Its main tasks are buffering the different network-side audio streams and compressing (audio/speech coding) the audio data, if desired.

To keep the actual network interface as simple as possible and to avoid excessive overhead due to additional headers, the Windows Socket interface is used to transmit the data through the network as a user datagram protocol (UDP) packet. This interface deals with all protocol tasks and requires only a port number and the IP address of the receiving client. At the receiving client the different audio modes (sample rate, compression mode) are distinguished by the block size of the incoming data. To detect missing or repeated packets, a modulo 256 counter is added at the beginning of the actual audio data and is incremented each time a new packet is sent.

The socket buffer is basically a FIFO (first in, first out) buffer which acts as a cache between the clients. It deals with three tasks: synchronization of data blocks, adjustment to different block sizes, and correcting for buffer underflow because of network problems. The latter two points are effects of the network and cannot be predicted whereas the first one is caused by the client itself. Since two clients are in general not synchronized, there is a small offset between the sample rates of the sound cards. As a result, one client transmits more packets in a certain time period than the other which results in a permanent drift in the FIFO buffer causing over- or under-runs. In this implementation, we simply detect these errors in the buffer and delete extra packets or insert packets filled with zeros to clear or refill the buffer. It is clear that these corrections will introduce audible effects, but if the buffer length is relative long and the sample rate offset is not too high, corrections are rarely necessary. On the other hand, enlarging the buffer will raise the overall audio latency of the connection. Therefore, the buffer size presents a trade-off between audio latency and immunity against synchronization problems. A more sophisticated solution would be to apply sample-rate correction to the incoming audio stream. In this case, however, the sample-rate offset must be estimated and the sample-rate conversion algorithm must be capable of converting to arbitrary sample rates.

The application uses an audio compression algorithm [9] that provides compression rates of 32, 24, or 16 kbit/s. Compression reduces the network load and makes a communication link possible even through slow analog modems.

2.3 THE ECHO CANCELER MODULE

The core echo canceler consists of a robust two-channel frequency-domain adaptive algorithm, a pseudo-coherence-based double-talk detector, and a residual echo suppression unit. The choice of this solution is based on the need for a low-complexity algorithm, as compared to a time-domain solution, as well as the need to handle the problem of slow convergence in the two-channel case. This could be achieved with a subband solution as in [10], however, a more complicated adaptive algorithm (fast recursive least squares) would be required. The frequency-domain solution has been shown to provide a reasonable trade-off between complexity and performance [11, 5]. The specific problem of stereophonic echo cancellation, i.e., the nonuniqueness problem, is handled by nonlinear distortion as described in [12]. Details regarding the adaptive algorithm and double-talk detector are presented in the next sections. The suppression algorithm attenuates the residual echo $e(n)$ (Fig. 7.1) depending on the actual amount of echo cancellation. The adaptation of the attenuation is based on voice activity detection decisions and an echo-return-loss measurement. All the measurements are based on the envelopes of the speech signals and the noise.

If the dynamic range of the received signal is near full range, unmodeled nonlinearities in the echo path will likely be excited. Therefore, compression of the incoming far-end signal's amplitude is done.

Since the analog input signal for the sound card must be amplified with analog circuits prior to analog-to-digital (A/D) conversion, it is most likely that the captured signal carries an unwanted DC component. This DC offset can lead to performance degradation or even to instability of the adaptive algorithm and must therefore be removed with a high-pass filter.

3. ALGORITHMS OF THE ECHO CANCELER MODULE

In this section, the robust two-channel frequency-domain adaptive algorithm and a frequency-domain-based double-talk detector are presented [5]. Furthermore, a sub-band noise and residual echo suppression structure is outlined.

For the stereo case we have two incoming transmission room signals, $x_p(n)$, $p = 1, 2$, where the input excitation vectors are defined as

$$\begin{aligned} \mathbf{x}_p(n) &= \begin{bmatrix} x_p(n) & x_p(n-1) & \cdots & x_p(n-L+1) \end{bmatrix}^T, \\ & \quad p = 1, 2. \end{aligned}$$

The error signal at time n between one arbitrary microphone output $y(n)$ in the receiving room and its estimate is

$$e(n) = y(n) - \sum_{p=1}^{2} \hat{y}_p(n) = y(n) - \sum_{p=1}^{2} \hat{\mathbf{h}}_p^T \mathbf{x}_p(n),$$

$$\hat{\mathbf{h}}_p = \begin{bmatrix} \hat{h}_{p,0} & \hat{h}_{p,1} & \cdots & \hat{h}_{p,L-1} \end{bmatrix}^T, \; p = 1, 2.$$

In general, we have two microphones in the receiving room, i.e. two return channels, and thus four adaptive filters. For simplicity in the derivation, only one return channel is considered; the derivation is similar for the other channel. Furthermore, the following block signals are defined:

$$\mathbf{e}(m) = [e(mL) \cdots e(mL + L - 1)]^T, \quad (7.1)$$

$$\mathbf{y}(m) = [y(mL) \cdots y(mL + L - 1)]^T, \quad (7.2)$$

where the block length is chosen to be equal to the length of the adaptive filter L and m is the block time index.

3.1 ADAPTIVE FILTER ALGORITHM

As with time-domain adaptive filter algorithms, the derivation begins with forming a criterion that is minimized with respect to the filter coefficients. Most commonly, the choice is a quadratic criterion that corresponds to a maximum likelihood estimator when the underlying noise distribution is Gaussian. Here, a maximum likelihood criterion derived from a non-Gaussian noise assumption is used. Modeling the noise with a probability density function (PDF) having a tail that is heavier than the Gaussian PDF gives a non-quadratic function to minimize, which results in an outlier-robust algorithm [13]. The following criterion is used:

$$J(\underline{\hat{\mathbf{h}}}) = \sum_{n=mL}^{mL+L-1} \rho\left[\frac{|e(n)|}{s}\right], \quad (7.3)$$

where $\rho\left[\cdot\right]$ is a convex function and s is a real positive scale factor as discussed below.

Minimizing (7.3) yields the robust two-channel frequency-domain adaptive algorithm [5, 14]:

$$\begin{aligned}
\underline{\mathbf{e}}(m) &= \underline{\mathbf{y}}(m) - \mathbf{G}^{01}\mathbf{D}(m)\underline{\hat{\mathbf{h}}}(m), && (7.4)\\
\mathbf{S}'(m) &= \lambda\mathbf{S}'(m-1) + (1-\lambda)\,\mathbf{D}^{H}(m)\mathbf{D}(m), && (7.5)\\
\mathbf{K}(m) &= \mathbf{G}^{10}\mathbf{S}'^{-1}(m)\mathbf{D}^{H}(m), && (7.6)\\
\underline{\hat{\mathbf{h}}}(m) &= \underline{\hat{\mathbf{h}}}(m-1) + \frac{2\mu' s(m)}{\psi'_{\mathrm{min}}}\mathbf{K}(m)\mathbf{F}\psi\left[\mathbf{e}(m)\right], && (7.7)\\
\psi'_{\mathrm{min}}(m) &= \max\left\{\min_{0\le l\le L-1}\left\{\psi'\left[\frac{|e(mL+l)|}{s}\right]\right\}, 0.5\right\}, &&\\
\psi\left[\mathbf{e}(m)\right] &= \begin{bmatrix} \mathbf{0}_{L\times 1} \\ \\ \psi\left[\dfrac{e(mL)}{s}\right] \\ \vdots \\ \psi\left[\dfrac{e(mL+L-1)}{s}\right] \end{bmatrix}, && (7.8)\\
\psi(z) &= \rho'(z) = \max\left[\min\{z, k_0\}, -k_0\right], && (7.9)\\
s(m+1) &= \lambda_{\mathrm{s}} s(m) &&\\
&\quad + (1-\lambda_{\mathrm{s}})\frac{s(m)}{L\beta}\sum_{n=mL}^{mL+L-1}\psi\left[\frac{|e(n)|}{s(m)}\right] && (7.10)
\end{aligned}$$

where $\mathbf{F}$ is the Fourier matrix and

$$\begin{aligned}
\underline{\hat{\mathbf{h}}}_p &= \mathbf{F}\begin{bmatrix} \hat{\mathbf{h}}_p \\ \mathbf{0}_{L\times 1} \end{bmatrix}, \ \underline{\hat{\mathbf{h}}} = \left[\underline{\hat{\mathbf{h}}}_1^T \ \underline{\hat{\mathbf{h}}}_2^T\right]^T,\\
\underline{\mathbf{y}}(m) &= \mathbf{F}\begin{bmatrix} \mathbf{0}_{L\times 1} \\ \mathbf{y}(m) \end{bmatrix},\\
\mathbf{G}^{01} &= \mathbf{F}\mathbf{W}^{01}\mathbf{F}^{-1},\\
\mathbf{W}^{01} &= \begin{bmatrix} \mathbf{0}_{L\times L} & \mathbf{0}_{L\times L} \\ \mathbf{0}_{L\times L} & \mathbf{I}_{L\times L} \end{bmatrix},\\
\mathbf{G}^{10} &= \mathbf{F}\mathbf{W}^{10}\mathbf{F}^{-1},
\end{aligned}$$

$$\begin{aligned}
\mathbf{W}^{10} &= \begin{bmatrix} \mathbf{I}_{L\times L} & \mathbf{0}_{L\times L} \\ \mathbf{0}_{L\times L} & \mathbf{0}_{L\times L} \end{bmatrix}, \\
\mathbf{D}_p(m) &= \operatorname{diag}\left\{ \mathbf{F} \begin{bmatrix} x_p(mL-L) \\ \vdots \\ x_p(mL+L-1) \end{bmatrix} \right\}, \\
\mathbf{D}(m) &= [\mathbf{D}_1(m)\,\mathbf{D}_2(m)]\,, && (7.11) \\
\mu' &= \mu(1-\lambda),\ (\mu,\ \lambda,\ \lambda_s) \in [\,0,\ 1\,]\,. && (7.12)
\end{aligned}$$

3.1.1 Double-talk Detection. A frequency-domain computation scheme for the normalized cross-correlation double-talk test statistic is presented in [5]. The decision variable is given by

$$\xi = \sqrt{\mathbf{s}^H \left(\sigma_y^2 \mathbf{S}\right)^{-1} \mathbf{s}} = \|\left(\sigma_y^2 \mathbf{S}\right)^{-1/2} \mathbf{s}\|_2 = \|\mathbf{c}_{xy}\|_2, \qquad (7.13)$$

where

$$\mathbf{S} = E\left[\mathbf{D}^H(m)\mathbf{G}^{01}\mathbf{D}(m)\right], \qquad (7.14)$$

is the spectral matrix of the transmission room signal, and $\mathbf{s} = E\left[\mathbf{D}^H(m)\underline{\mathbf{y}}(m)\right]$ is a cross-spectral vector between the transmission room and receiving room signals. The vector $\mathbf{c}_{xy}$ is referred to as the pseudo-coherence vector [15]. Looking at (7.13), each cross-spectrum bin of $\mathbf{s}$ is normalized by the corresponding spectrum of the input signal, $\mathbf{x}$. What differentiates (7.13) from being the true coherence is that it is not normalized by the corresponding spectrum of the output signal y but by the total power of the output signal, σ_y^2. A practical double-talk detection statistic can now be realized by using estimated quantities in (7.13) and slightly re-writing the numerator,

$$\xi^2(m) = \frac{\mathbf{s}^H(m)\mathbf{S}^{-1}(m)\mathbf{s}(m)}{\sigma_y^2(m)} = \frac{\mathbf{s}^H(m)\hat{\underline{\mathbf{h}}}_{\mathrm{b}}(m)}{\sigma_y^2(m)}, \qquad (7.15)$$

where the square of the statistic is used for simplicity, and $\hat{\underline{\mathbf{h}}}_{\mathrm{b}}(m) = \mathbf{S}^{-1}(m)\mathbf{s}(m)$ is defined as an equivalent "background" filter. The decision variable is obtained by using a separate filter for the DTD (not to be confused with the "foreground" estimate of the echo canceler). The nomenclature "background" and "foreground" is borrowed from a double-talk method called the two-path method [16]. In some literature, the background filter is called the shadow filter [17]. Estimates of the quantities in (7.15) are given by

$$\begin{aligned}
\mathbf{s}(m) &= \lambda_{\mathrm{b}}\mathbf{s}(m-1) + (1-\lambda_{\mathrm{b}})\mathbf{D}^H(m)\underline{\mathbf{y}}(m), && (7.16) \\
\mathbf{S}(m) &= \lambda_{\mathrm{b}}\mathbf{S}(m-1) + (1-\lambda_{\mathrm{b}})\mathbf{D}^H(m)\mathbf{G}^{01}\mathbf{D}(m), && (7.17) \\
\sigma_y^2(m) &= \lambda_{\mathrm{b}}\sigma_y^2(m-1) + (1-\lambda_{\mathrm{b}})\underline{\mathbf{y}}^H(m)\underline{\mathbf{y}}(m). && (7.18)
\end{aligned}$$

As a simplification, the background echo path estimate $\underline{\hat{\mathbf{h}}}_{\mathrm{b}}(m)$ in (7.15) can be computed adaptively as

$$\begin{aligned}\underline{\hat{\mathbf{h}}}_{\mathrm{b}}(m) &= \underline{\hat{\mathbf{h}}}_{\mathrm{b}}(m-1) + \mu_{\mathrm{b}}\mathbf{K}(m)\underline{\mathbf{e}}_{\mathrm{b}}(m), && (7.19)\\ \mu_{\mathrm{b}} &= (1-\lambda_{\mathrm{b}}), && (7.20)\end{aligned}$$

where $\mathbf{S}'(m)$ is defined in (7.5) and

$$\underline{\mathbf{e}}_{\mathrm{b}}(m) = \underline{\mathbf{y}}(m) - \mathbf{G}^{01}\mathbf{D}(m)\underline{\hat{\mathbf{h}}}_{\mathrm{b}}(m-1). \qquad (7.21)$$

The background estimate should be adapted with a smaller forgetting factor, λ_{b}, than that of the foreground filter, λ. By this choice, the DTD detects double-talk fast and alerts the foreground filter before it diverges. Table 7.1 summarizes the robust two-channel frequency-domain adaptive filter combined with the DTD.

4. RESIDUAL ECHO AND NOISE SUPPRESSION

The adaptive filter is the key component in echo cancellation. However, in the acoustic application, a linear echo canceler is often inadequate to provide sufficient cancellation such that the residual echo is inaudible. This is particularly so in cases where there is a nonlinear imperfection in the echo path or the echo path changes because of motion in the acoustic path. Unfortunately, suppression impairs the "duplexness" (near-end speech transparentness) of the echo cancellation system. The problem presented by the use of an echo suppressor is how to trade off duplexness for satisfactory performance. Furthermore, it is often advantageous to improve the perceived quality by reducing the ambient noise with a noise suppressor before transmitting the microphone signal. These two functions can be implemented jointly. The system of a (mono) echo canceler, residual echo and noise suppressor is shown in Figure 7.3. The residual echo and noise suppressor is denoted by the time varying system function $g(n)$. Finally, artifacts may be introduced by the suppression methods so it is customary to mask these distortions by adding "comfort noise," $\omega(n)$. The combined suppression and comfort noise injection operation can be described as

$$e_s(n) = g(n) * e(n) + \omega(n). \qquad (7.22)$$

In general, $g(n)$ can be a time varying filter function of arbitrary order. Here, subband (or frequency-domain) implementations are used to allow independent processing of each frequency band as described in [18]. This choice casts the suppressor as a scalar attenuator, *G,* in each frequency subband[2].

The objective in this section is to present the achievable cancellation performance of the adaptive algorithm of the AEC (the results are also applicable to the algorithm presented in the previous section as well as any general gradient-based adaptive algorithm). Based on this result, relations are derived between

Table 7.1 The two-channel PC double-talk detector and echo canceler. The parameter ϱ bounds the maximum allowed coherence between the channels.

Spectral estimation

$$\begin{aligned}
\mathbf{S}'_{p,q}(m) &= \lambda \mathbf{S}'_{p,q}(m-1) \\
&\quad + (1-\lambda)\,\mathbf{D}_p^H(m)\mathbf{D}_q(m),\ p,q = 1,2 \\
\tilde{\mathbf{S}}_{p,p}(m) &= \mathbf{S}'_{p,p}(m) + \mathrm{diag}\{\delta_{p,0}\ldots\delta_{p,2L-1}\},\ p = 1,2 \\
|\mathbf{\Gamma}^2(m)| &= \left[\tilde{\mathbf{S}}_{1,1}(m)\tilde{\mathbf{S}}_{2,2}(m)\right]^{-1} \mathbf{S}'_{2,1}(m)\mathbf{S}'_{1,2}(m) \\
\mathbf{S}_p(m) &= \tilde{\mathbf{S}}_{p,p}(m)\times \\
&\quad \left[\mathbf{I}_{2L\times 2L} - \varrho^2|\mathbf{\Gamma}^2(m)|\right],\ p,q = 1,2 \\
\mathbf{K}_1(m) &= \mathbf{S}_1^{-1}(m)\left[\mathbf{D}_1^*(m) - \varrho\mathbf{S}'_{1,2}(m)\tilde{\mathbf{S}}_{2,2}^{-1}(m)\,\mathbf{D}_2^*(m)\right] \\
\mathbf{K}_2(m) &= \mathbf{S}_2^{-1}(m)\left[\mathbf{D}_2^*(m) - \varrho\mathbf{S}'_{2,1}(m)\tilde{\mathbf{S}}_{1,1}^{-1}(m)\,\mathbf{D}_1^*(m)\right]
\end{aligned}$$

Double-talk detector (Background filter)

$$\begin{aligned}
\underline{\mathbf{e}}_{\mathrm{b}}(m) &= \underline{\mathbf{y}}(m) - \mathbf{G}^{01}\mathbf{D}(m)\hat{\underline{\mathbf{h}}}_{\mathrm{b}}(m-1) \\
\hat{\underline{\mathbf{h}}}_{\mathrm{b},p}(m) &= \hat{\underline{\mathbf{h}}}_{\mathrm{b},p}(m-1) + (1-\lambda_{\mathrm{b}})\mathbf{K}_p(m)\underline{\mathbf{e}}_{\mathrm{b}}(m),\ p = 1,2 \\
\mathbf{s}(m) &= \lambda_{\mathrm{b}}\mathbf{s}(m-1) + (1-\lambda_{\mathrm{b}})\,\mathbf{D}^H(m)\underline{\mathbf{y}}(m) \\
\eta^2(m) &= \left[\hat{\underline{\mathbf{h}}}_{\mathrm{b},1}^H(m)\ \hat{\underline{\mathbf{h}}}_{\mathrm{b},2}^H(m)\right]\mathbf{s}(m) \\
\sigma_{\underline{\mathrm{y}}}^2(m) &= \lambda_{\mathrm{b}}\sigma_{\underline{\mathrm{y}}}^2(m-1) + (1-\lambda_{\mathrm{b}})\underline{\mathbf{y}}^H(m)\underline{\mathbf{y}}(m) \\
\xi(m) &= \eta(m)/\sigma_{\underline{\mathrm{y}}}(m) < T,\ \Rightarrow \mu' = 0 \\
\xi(m) &= \eta(m)/\sigma_{\underline{\mathrm{y}}}(m) \geq T,\ \Rightarrow \mu' = \mu(1-\lambda)
\end{aligned}$$

Echo canceler (Foreground filter)

$$\begin{aligned}
\mathbf{e}(m) &= \mathbf{y}(m) - \mathbf{W}^{01}\mathbf{F}^{-1}\mathbf{D}(m)\hat{\underline{\mathbf{h}}}(m-1) \\
\hat{\underline{\mathbf{h}}}_p(m) &= \hat{\underline{\mathbf{h}}}_p(m-1) + \frac{2\mu' s(m)}{\psi'_{\min}}\mathbf{K}_p(m)\mathbf{F}\psi\left[\mathbf{e}(m)\right],\ p = 1,2 \\
s(m+1) &= \lambda_{\mathrm{s}} s(m) + (1-\lambda_{\mathrm{s}})\frac{s(m)}{L\beta}\sum_{n=mL}^{mL+L-1}\psi\left[\frac{|e(n)|}{s(m)}\right]
\end{aligned}$$

adaptive algorithm parameters, and the residual echo suppression required using perceptual knowledge of the human auditory system. An outline of a joint noise and residual echo suppression is then given.

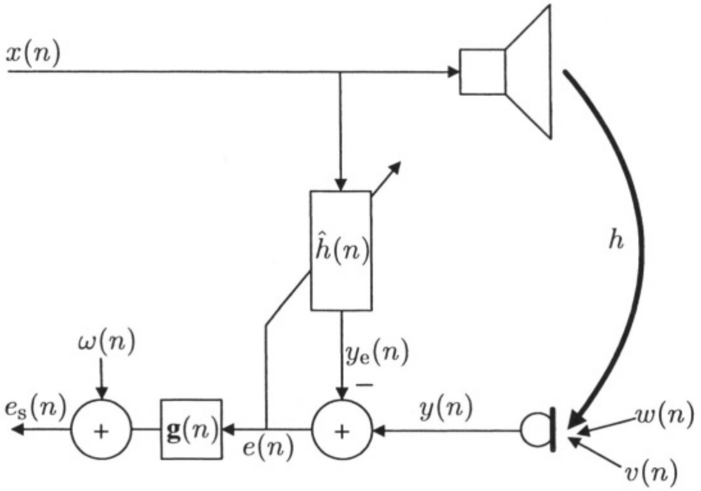


Figure 7.3 Block diagram of the generic (mono) echo canceler, echo path, and noise/residual echo suppression filter.

4.1 MASKING THRESHOLD FOR RESIDUAL ECHO IN NOISE

Guiding the design of any speakerphone algorithm is the necessity to attenuate the residual echo to make it inaudible under the present noise situation. To address this requirement, an understanding of the masking effects of speech in noise is needed. Unfortunately, there is no rigid investigation for this specific case. However, cases in which noise masks a tone or vice versa have been thoroughly investigated [19, 20]. For the relevant case of a pure tone in narrow-band noise (equal or smaller than one critical bandwidth), the tone becomes inaudible if it is 3 dB or more below the noise level [19]. Speech or residual echo (filtered speech) is highly audible at this speech-to-noise ratio when average power level of speech is considered. However, speech compared to tones has a much higher peak-to-standard deviation ratio, i.e., the crest-factor of speech c_s is larger than that of tones. Denoting the variance of speech by σ_s^2, if instead of using the average speech power one compares the "equivalent tone power of speech" (ETP)

$$\mathrm{ETP} = \frac{\sigma_s^2 c_s^2}{2}, \qquad (7.23)$$

with the noise variance σ_w^2, the residual echo (speech) should be inaudible if

$$\frac{\mathrm{ETP}}{\sigma_w^2} \le 10^{-3/10}$$

or

$$\frac{\sigma_s^2}{\sigma_w^2} \le \frac{2}{c_s^2} 10^{-3/10}. \tag{7.24}$$

Clean speech has a crest-factor of around 20 [21] but the authors' experience is that residual echo has a lower value around 10. This means that (7.24) is in the range –20 dB ($c_s = 10$) to –26 dB ($c_s = 20$). We have performed listening experiments with residual echo in various levels of white noise that have confirmed this approximate required ratio (–20 dB) for inaudibility.

4.2 ANALYSIS OF ECHO SUPPRESSION REQUIREMENTS

In this section, the performance and requirements of the echo canceler and residual echo suppressor are quantified. Assume that the adaptive algorithm of the echo canceler has the generic update equation

$$\hat{\mathbf{h}}(m) = \hat{\mathbf{h}}(m-1) - 2\mu(1-\lambda)\Delta\hat{\mathbf{h}}(m), \tag{7.25}$$

where $\hat{\mathbf{h}}$ is the filter vector of length L and $\Delta\hat{\mathbf{h}}(m)$ is the gradient of the criterion that we seek to minimize. The performance, i.e. accuracy, of the estimate is solely determined by the step-size factor $2\mu(1-\lambda)$ and the echo-to-background noise ratio (EBR). Analyses show [5, Chapt. 9] that the performance measured as excess MSE is given by

$$\text{ex. MSE} = \frac{\sigma_{e-w}^2}{\sigma_{y-w}^2} = \frac{\mu(1-\lambda)}{\mathrm{EBR}} \approx \frac{\mu}{K_0 \cdot \mathrm{EBR}}, \tag{7.26}$$

where σ_{e-w}^2 is the residual echo and σ_{y-w}^2 is the (uncancelled) echo. The parameter $K_0 > 0$ is related to forgetting factor λ as

$$\lambda = \left(1 - \frac{1}{K_0 L}\right)^L. \tag{7.27}$$

Equation (7.26) is also valid for the traditional normalized least-mean-square algorithm (NLMS) [22] if $K_0 = 2$ [23]. Hence, after initial convergence, the performance of most echo cancelers can be quantified by the parameters $\mu \in (0, 1]$ and $K_0 > 1$.

The ex. MSE only assesses the cancellation performance and not the audible residual echo (perceived performance). The perceived performance is quantified by the residual echo-to-noise ratio (RENR). The relation between the step-size parameters and RENR is found using the ex. MSE as

$$\text{ex. MSE} = \frac{\sigma_{e-w}^2}{\sigma_{y-w}^2} = \frac{\mu}{K_0 \cdot \text{EBR}} = \frac{\mu\sigma_w^2}{K_0\sigma_{y-w}^2}. \tag{7.28}$$

Reordering (7.28) gives

$$\text{RENR} = \frac{\sigma_{e-w}^2}{\sigma_w^2} = \frac{\mu}{K_0}. \tag{7.29}$$

It is also worthwhile to note that the RENR can be approximately related to EBR and the misalignment (MIS) of the adaptive filter

$$\text{RENR} = \frac{\sigma_{e-w}^2}{\sigma_w^2} = \frac{\sigma_x^2\|\mathbf{h}\|^2}{\sigma_w^2} \cdot \frac{\|\mathbf{h}-\hat{\mathbf{h}}\|^2}{\|\mathbf{h}\|^2} \approx \text{EBR} \cdot \text{MIS}. \tag{7.30}$$

The overall perceived performance of the AEC and suppression system is given by the total residual echo-to-noise ratio (RENR_0). To capture this quantity in a simple expression, the suppression function $g(n)$ in Fig. 7.3 is constrained to be a scalar, G. Also, assume that the echo and noise have constant spectra over frequency. After suppression, the desired total noise variance is equal to a factor $\gamma_\omega^2 G_s^2 \leq 1$ of the original noise variance. The natural noise is attenuated by G_s^2 and the amount of comfort noise added is controlled by a factor $\gamma_\omega^2 > 0$. This leads to

$$G^2\sigma_v^2 + \sigma_\omega^2 = G_s^2\gamma_\omega^2\sigma_w^2, \tag{7.31}$$

and the overall residual echo-to-noise ratio *after* suppression and addition of comfort noise, RENR_0, then becomes

$$\text{RENR}_0 = \frac{G^2\sigma_{e-w}^2}{G^2\sigma_v^2 + \sigma_\omega^2} = \frac{G^2\sigma_{e-w}^2}{G_s^2\gamma_\omega^2\sigma_w^2} = \text{RENR}\frac{G_{\text{ex}}^2}{\gamma_\omega^2}, \tag{7.32}$$

where $G_{\text{ex}} = G/G_s$ is the extra residual echo suppression. Reordering (7.32) results in

$$G_{\text{ex}}^2 = \gamma_\omega^2 \frac{\text{RENR}_0}{\text{RENR}}. \tag{7.33}$$

To summarize:

- The required suppression (G_{ex}) to achieve inaudible residual echo is given by (7.33).

- RENR after convergence is given by the setting of the echo canceler parameter (7.29) or in general (when AEC has not converged) by (7.30).
- RENR_0 is given by psycho-acoustic considerations (7.24) and experimental results.
- γ_ω defines the final noise level in the system.

For practical convergence rates, i.e. for practical choices of μ and K_0, RENR is approximately –10 dB. However, in (7.30), RENR may become larger than EBR (normally is 10 – 20 dB) when a severe echo path change occurs (MIS $\approx 1 - 2$ in this case). Under this condition, with $\gamma_\omega = 1$ and $\mathrm{RENR}_0 = -26$ dB [from (7.24)], the residual echo attenuation required is

$$G_{\mathrm{ex}}^2(\mathrm{in\,dB}) \quad = \quad 10\log_{10}\left(\gamma_\omega^2 \frac{\mathrm{RENR}_0}{\mathrm{RENR}}\right) \approx -26 - 20 = -46\,\mathrm{dB}(7.34)$$

4.3 NOISE AND RESIDUAL ECHO SUPPRESSION

Based on the ideas above, residual echo and noise suppression algorithms are implemented in a subband structure. Voice activity detection is performed in each subband of the echo and estimated echo signals. Depending on voice activity, statistics from these estimates are used for computation of a gain function which is then applied to the residual echo. The algorithm distinguishes between situations of far-end speech only and near-end speech only. In the case of double-talk the residual echo (and near-end speech) is attenuated using a Wiener filter structure.

5. SIMULATIONS

For these two-channel simulations, two-channel speech recordings were used [10, 24]. The sampling rate is 16 kHz. For transmission room speech, stereo recordings from a male talker are used. The transmission room speech is pre-processed with a nonlinearity before being emitted in the receiving room. By doing so, the echo canceler converges to a stable, unique solution. The nonlinearity used is [12]:

$$\begin{aligned} x_1' &= x_1 + \frac{\alpha}{2}\left[x_1 + |x_1|\right], \\ x_2' &= x_2 + \frac{\alpha}{2}\left[x_2 - |x_2|\right], \end{aligned}$$

where the subscript 1 or 2 denotes either left or right channel, respectively. Thus, the positive half-wave is added to the left channel and the negative to the right. The distortion parameter of the nonlinearity is α. Adaptive filter parameters are $L = 1024$ (64 ms), $\lambda = [1 - 1/(3L)]^{L/o}$, $o = 4$ (o is the overlapping

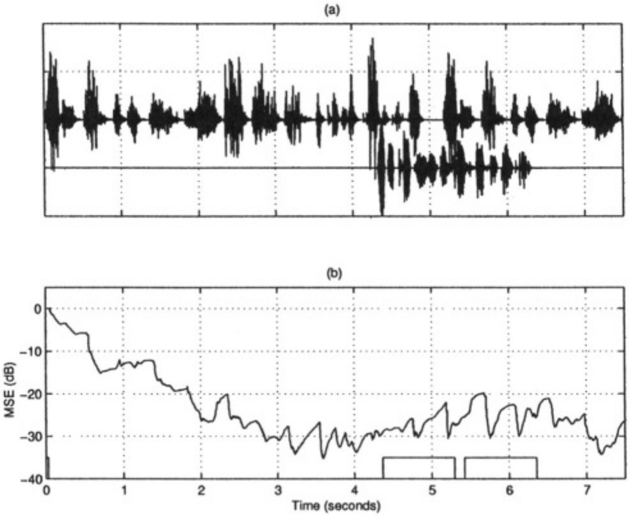


Figure 7.4 Effects of double-talk in terms of mean-square error of the two-channel robust frequency-domain algorithm with the pseudo-coherence-based DTD. (a) Left-channel transmission room speech (upper), receiving room speech (lower). (b) Results when double-talk is present between 4.3 and 6.2 s. The rectangles in the bottom of the figures indicate where double-talk has been detected.

factor), $\mu = 0.5$, $k_0 = 1.5$, $\underline{\hat{\mathbf{h}}}(0) = \mathbf{0}$, $\varrho = 0.99$ (relaxation parameter, see Table 7.1). DTD parameters are $\lambda_b = [1 - 2/(3L)]^{L/o}$, ENR = 1000 (30dB), $T = 0.92$. Table 7.1 presents the modified system of two-channel PC DTD and frequency-domain algorithm used in the simulations.

Figures 7.4 and 7.5 show the behavior of the system during double-talk and after an echo path change by means of its mean-square error (MSE). The mean-square error is defined as

$$\mathrm{MSE}(n) = \frac{\mathrm{LPF}\left\{[e(n) - v(n)]^2\right\}}{\mathrm{LPF}\left\{[y(n) - v(n)]^2\right\}}.$$

From these curves, it can be concluded that the system is robust to double-talk, yet at the same time shows rapid convergence after echo path changes because of proper behavior of the double-talk detector in the different situations. Figure 7.6 shows the MSE after echo path change (same as in Fig. 7.5), and the residual echo suppressor active. For the gain change, Fig. 7.6(b), there is a slight increase in returned residual echo.

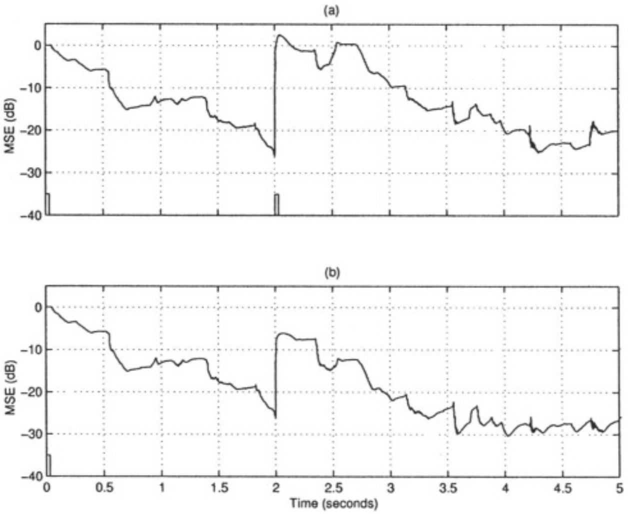


Figure 7.5 . Effects of echo path changes in terms of mean-square error of the two-channel robust frequency-domain algorithm with the pseudo-coherence-based DTD (no double-talk present). The echo paths in the receiving room change as: (a) Time delay of 5 samples at 2 s. (b) 6-dB increase of echo path gain at 2 s. The rectangles in the bottom of the figures indicate where double-talk has been detected (in this case, they are false alarms).

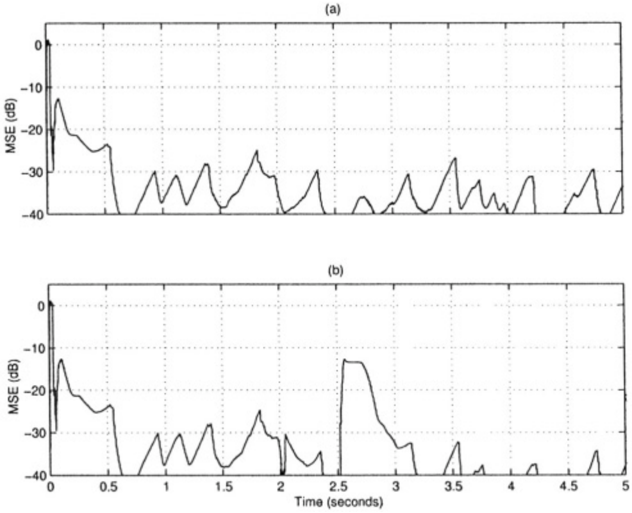


Figure 7.6 Data and conditions as in Fig. 7.5 except the residual echo suppressor is active. The echo paths in the receiving room change as: (a) Time delay of 5 samples at 2 s. (b) 6-dB increase of echo path gain at 2 s.

6. REAL-TIME TESTS WITH DIFFERENT MODES OF OPERATION

The authors have tested and evaluated the system in various environments and operation modes. Point-to-point communication in full stereo or mono has been made in an office environment, using a desktop or a laptop computer, and in a larger conference room, where the loudspeakers and microphones are far apart. Multi-point communication has been performed using a mix of laptop and desktop machines.

6.1 POINT-TO-POINT COMMUNICATION

In point-to-point communication, two clients (PCs) are directly connected to each other through a network. Tests with typical office environments showed that a sufficient impulse response length is 64 ms for achieving good echo cancellation. It is also possible to use the shorter span of 32 ms, but the communications quality decreases somewhat because the suppressor must act more aggressively to reduce residual echo.

In a conference-room situation, the audio components are physically distributed, hence the distance between the microphone and loudspeakers is increased. Furthermore, the room is usually larger than a regular office. In this case, a 32 ms filter will result in poor echo cancellation, especially in stereo mode where the tail effects become severe [12]. The minimum length of the estimated filter that worked well was 64 ms, but an audible improvement was achieved with 128 ms.

What characterizes a laptop audio system is the close arrangement of the microphone and the loudspeakers. Moreover, the loudspeakers are often of poor quality. There are also significant levels of noise originating from computer components (e.g., hard drive) located close to the microphone. In this situation, it is not possible to improve echo cancellation by simply increasing the impulse response length. A good choice is a short filter of length 32 ms. Other serious problems include the nonlinearity of the loudspeaker and the acoustic resonance of the laptop case, both of which cannot be properly modeled with a linear FIR filter of finite length. Also, the keyboard is often between the microphone and the loudspeaker and, if it is used, the impulse response rapidly changes all the time. To achieve good performance in this environment, accurate suppression is required.

6.2 MULTI-POINT COMMUNICATION

The conference server, also designed by the authors, is an audio bridge that connects many clients and creates synthesized stereo or 3D-audio streams. Each user that connects to the server will hear all other clients connected to the

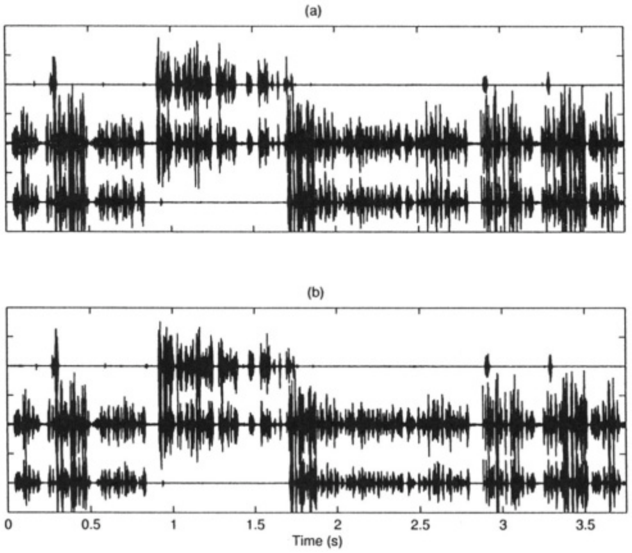


Figure 7.7 Recorded speech from first transatlantic teleconference in stereo. (a) Right channel. (b) Left channel. In order from top to bottom: Transmission speech (upper), echo and receiving speech (middle), and processed echo and receiving speech (lower).

same server. Because returning echoes are summed at the server, this situation places tougher requirements on echo cancellation. Furthermore, because of the nonuniqueness problem, the most difficult situation for the echo canceler to handle is the synthetic stereo/3D-audio case. Therefore, all clients have to adjust the cancellation parameters (i.e. impulse response length, suppression parameters etc.) and the audio system very carefully. With synthesized stereo and four clients, the positions of the different speakers are well distinguished. With 3D-audio, the audio distribution can be improved.

6.3 TRANSATLANTIC TELECONFERENCE IN STEREO

The WinEC system has been successfully used for transatlantic point-to-point communication over commercial networks. We performed the first stereo conference on January 31, 2003 between Chatham, New Jersey, USA and Darmstadt, Germany. In this test we used mono and stereo mode and audio in both coded (32/64 kbit/s) and uncoded (256/512 kbit/s) formats. Audio quality was outstanding (i.e. significantly better that regular phone toll quality) and very few network problems were experienced. The spatial properties of the audio really creates an increased sense of presence. A clip of the transmission speech, echo and receiving speech, and return speech are shown in Fig. 7.7. The com-

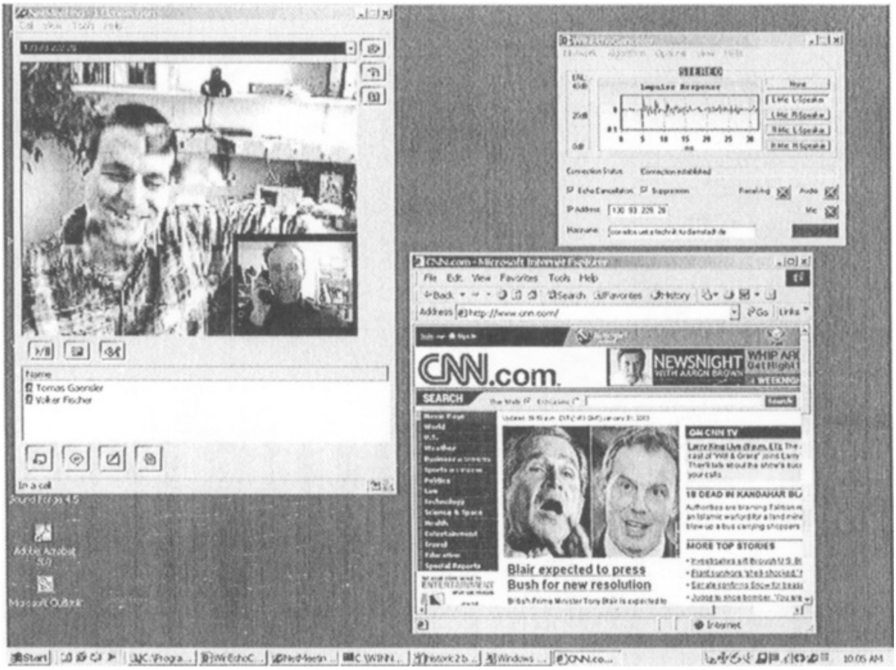


Figure 7.8 Computer screen with the WinEC application, Netmeeting (only used to show video of the participants, V. Fischer and T. Gänsler. Netmeeting audio has been disabled), and Internet Explorer running concurrently. While the latter participant is communicating hands-free with V. Fischer, he is also talking over a PSTN line with G. Elko, therefore the handset.

puter screen with our WinEC application, Microsoft's Netmeeting, and Internet Explorer is shown in Fig 7.8. We believe that this experiment represents the first ever transatlantic PC-based hands-free full duplex stereo conference over commercial IP networks. Our experience from this and subsequent evaluation sessions is that this technology is ready for commercialization.

7. DISCUSSION

In this chapter, we have described an implementation of a flexible stereophonic acoustic echo canceler. This implementation runs natively under Microsoft Windows on a PC. The major obstacle with such a scheme is the synchronization problem of the input and output audio streams of the sound card. Without proper synchronization, good cancellation cannot be maintained.

Evaluation of the echo canceler has shown that it achieves the theoretical bounds on performance (echo attenuation) which in general is approximately 5 dB below the room noise level (algorithm parameter dependent). This performance is valid for an echo-to-noise ratio down to about 35 dB. In practice, one cannot expect more cancellation because of the linear model mismatch, non-stationary room responses, and unmodeled tails of the responses. Attenuation of 20 to 35 dB is not sufficient since the round-trip delay of the system is fairly large, about 350 ms. This delay is mainly due to delay in the sound card and network interface and is a function of the involved buffers' lengths (assuming insignificant network delay). Because of this, residual echo suppression is required and has been implemented.

The application supports mono, natural full stereo, and synthetic stereo hands-free communication. Multi-point communication can be done in mono or synthetic stereo/3D-audio mode.

Notes

1. In a real-life situation, we need an echo canceler for the "transmission room" as well. However, for simplicity, we chose to exclude it in the figure.
2. The analysis is also valid for the time-domain function of order zero (scalar).

References

[1] B. H. Nitsch, "Real-time implementation of the exact block NLMS algorithm for acoustic echo control in hands-free telephone systems," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., chapter 4, pp. 68–80. Kluwer Academic Publishers, 2000.

[2] V. Fischer, T. Gänsler, E. J. Diethorn, and J. Benesty, "A software stereo scoustic echo canceler for Microsoft Windows," in *Proc. IWAENC,* 2001.

[3] T. Gänsler, J. Benesty, E. J. Diethorn, and V. Fischer, "Algorithm design of a stereophonic acoustic echo canceler system," in *Proc. IEEE ASSP Workshop Appls. Signal Processing Audio Acoustics,* pp. 179–182, 2001.

[4] T. Gänsler and J. Benesty, "Multichannel acoustic echo cancellation: what's new?," in *Proc. IWAENC,* 2001.

[5] J. Benesty, T. Gänsler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo cancellation,* Springer-Verlag, Berlin, 2001.

[6] J. Benesty, D. R. Morgan, J. L. Hall, and M. M. Sondhi, "Synthesized stereo combined with acoustic echo cancellation for desktop conferencing," *Bell Labs Tech. J.,* vol. 3, pp. 148–158, July-Sept. 1998.

[7] J. Chen, "3D audio and virtual acoustical environment synthesis," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., chapter 13, pp. 283–301. Kluwer Academic Publishers, 2000.

[8] Microsoft Corporation, "Microsoft developer network 6.0," www.msdn.com, 2000.

[9] S. A. Ramprashad, "A multimode transform predictive coder (MTPC) for speech and audio," in *IEEE Speech Coding Workshop,* 1999.

[10] P. Eneroth, S. L. Gay, T. Gänsler, and J. Benesty, "A real-time stereophonic acoustic subband echo canceler," in *Acoustic Signal Processing for Telecommunications,* S. L. Gay and J. Benesty, Eds., chapter 8, pp. 135–152. Kluwer Academic Publishers, 2000.

[11] P. Eneroth, J. Benesty, T. Gänsler, and S. L. Gay, "Comparison of different adaptive algorithms for stereophonic acoustic echo cancellation," in *Proc. EUSIPCO*, 2000, pp. 1835–1837.

[12] J. Benesty, D. R. Morgan, and M. M. Sondhi, "A better understanding and an improved solution to the specific problems of stereophonic acoustic echo cancellation," *IEEE Trans. Speech Audio Processing,* vol. 6, pp. 156–165, Mar. 1998.

[13] P. J. Huber, *Robust Statistics,* Wiley, New York, 1981.

[14] J. Benesty and D. R. Morgan, "Multi-channel frequency-domain adaptive filtering," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., chapter 7, pp. 121–133. Kluwer Academic Publishers, 2000.

[15] T. Gänsler and J. Benesty, "A frequency-domain double-talk detector based on a normalized cross-correlation vector," *Signal Processing,* vol. 81, pp. 1783–1787, Aug. 2001.

[16] K. Ochiai, T. Araseki, and T. Ogihara, "Echo canceler with two echo path models," *IEEE Trans. Commun.,* vol. 25, pp. 589–595, June 1977.

[17] A. Mader, H. Puder, and G. U. Schmidt, "Step-size control for acoustic echo cancellation filters – an overview," *Signal Processing,* vol. 80, pp. 1697–1719, Sept. 2000.

[18] E. J. Diethorn, "Subband noise reduction methods for speech enhancement," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., chapter 9, pp. 156–178. Kluwer Academic Publishers, 2000.

[19] E. Zwicker and H. Fastl, *Psycho-acoustics, Facts and Models*, Information Sciences. Springer, 2nd edition, 1999.

[20] H. Fletcher, *Speech and Hearing in Communication,* The Acoustical Society of America, 1995.

[21] *Transmission Systems for Communications,* Bell Laboratories, fifth edition, 1982.

[22] S. Haykin, *Adaptive Filter Theory,* Prentice-Hall, Englewood Cliffs, NJ, 1996.

[23] T. Gänsler, J. Benesty, and S. L. Gay, "Double-talk detection schemes for acoustic echo cancellation," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., chapter 5, pp. 81–97. Kluwer Academic Publishers, 2000.

[24] D. A. Berkley and J. L. Flanagan, "HuMaNet: an experimental human-machine communications network based on ISDN wideband audio," *AT&T Tech. J.,* vol. 69, pp. 87–99, Sept./Oct. 1990.

This page intentionally left blank

III

SOUND SOURCE TRACKING AND SEPARATION

This page intentionally left blank

Chapter 8

TIME DELAY ESTIMATION

Jingdong Chen
Bell Laboratories, Lucent Technologies
jingdong@research.bell-labs.com

Yiteng (Arden) Huang
Bell Laboratories, Lucent Technologies
arden@research.bell-labs.com

Jacob Benesty
Université du Québec, INRS-EMT
benesty@inrs-emt.uquebec.ca

Abstract Time delay estimation has been a research topic of significant practical importance in many fields (radar, sonar, seismology, geophysics, ultrasonics, hands-free communications, etc.). It is a first stage that feeds into subsequent processing blocks for identifying, localizing, and tracking radiating sources. This area has made considerable advances in the past few decades, and is continuing to progress, with an aim to create processors that are tolerant to both noise and reverberation. This chapter reviews some recently developed algorithms for time delay estimation. The emphasis is placed on their performance analysis and comparison in reverberant environments. In particular, algorithms reviewed include the generalized cross-correlation algorithm, the multichannel cross-correlation algorithm, and the blind channel identification technique based algorithms. Furthermore, their relations and improvements are also discussed. Experiments based on the data recorded in the Varechoic chamber at Bell Labs are provided to illustrate their performance differences.

Keywords: Time Delay Estimation, Time of Arrival Estimation, Time Difference of Arrival Estimation, Multipath Effect, Reverberation, Blind Channel Identification, Coherence Function, Cross-Correlation Function, Eigenvalue Decomposition, Multichannel, Adaptive Algorithms, LMS, Newton's Method

1. INTRODUCTION

Time delay estimation (TDE) has been an area of great research interest for many decades. It has plenty of applications in fields as diverse as radar, sonar, seismology, geophysics, ultrasonics, and communications for detecting, identifying, and localizing radiating sources.

Depending on the nature of its application, TDE can be dichotomized into two broad categories, namely, the time of arrival (TOA) estimation [1, 2, 3, 4] and the time difference of arrival (TDOA) estimation [5, 6]. The former aims at measuring the time delay between the transmission of a pulse signal and the reception of its echo, which is often of primary interest to an active system such as radar and active sonar; while the latter, as its name indicates, endeavors to determine the relative time difference of arrival between two spatially separated sensors, which is often of concern to a passive system such as passive sonars and microphone array systems. Though there exists intrinsic relationship between the TOA and TDOA estimation, their essential difference is literally profound. In the former case, the "clean" reference signal, i.e., the transmitted signal, is known, such that the time delay estimate can be obtained based on a single sensor generally using the matched filter approach. On the contrary, in the latter, no such explicit reference signal is available, and the delay estimate is often acquired by comparing the signals received at two (or more) spatially separated sensors. This chapter deals with the subject of time delay estimation, with emphasis on the time difference of arrival. From now on, we will make no distinction between TDE and TDOA estimation unless necessary.

Measuring TDOA among different channels is a fundamental approach to detecting, localizing, and tracking radiating sources. Over the past few decades, researchers have approached this challenging problem by exploiting different facets of the received signals. Some good reviews of such efforts can be found in [5, 7, 8, 9]. Fundamentally, the solutions to the problem can be categorized from the following points of view:

- The number of sources in the field, i.e., single-source TDE techniques [5] and multiple-source TDE techniques [10, 11].
- How the propagation condition is modeled, i.e., the ideal single-path propagation model [5], the multipath propagation model [12, 13, 14], and the convolutive reverberant model [15].
- What analysis tools are employed, e.g., generalized cross-correlation method [5, 16, 17, 18, 19, 20], higher-order statistics (HOS) based approaches [21, 22], and blind channel identification based algorithms [15, 23].
- How the delay estimate is updated, i.e., non-adaptive and adaptive [24, 25, 26, 27, 28] approaches.

The vast majority of existing TDE algorithms deal with a single source. While estimating the time delay of multiple targets is an important issue, this work focuses on the single-source scenario only. The normal practice in such a task involves identifying the maximum of the generalized cross-correlation (GCC) function of the outputs of two sensors. This so-called GCC method consists of two prefilters followed by a cross correlator. The prefilters operate on the observation sequences in the frequency domain to control the TDE performance and their transfer functions are chosen according to some criterion. Some prefilters are optimal in the sense that the estimation variance can approach the Cramèr-Rao lower bound [such as the maximum likelihood (ML) approach]. Some are sub-optimal but possess special properties, for example, the ability to deal more efficiently with noise. The GCC approach is well studied and can produce reasonable performance in the single-path propagation situation.

Recently, there has been a growing interest in exploring the TDE technique in a room acoustic environment where the time delay estimation becomes more complicated owing to the sophisticated mutlipath effect: in addition to the direct path, the source wavefront reaches the receiver after getting reflected off room boundaries such as walls and floors, and other objects in the room. This multipath effect introduces echoes and spectral distortion into the speech signal, which is termed reverberation. The GCC algorithm, and many other traditional methods tend to break down in such reverberant environments [29].

Much attention has been paid to combatting reverberation lately. Most of such efforts fall into two categories. The first is to use multiple (more than two) sensors and take advantage of the redundancy to improve the TDE performance. Two examples are the Ferguson's method [30] and consistency-based method [31]. Ferguson's approach is an extension of the GCC algorithm in which an array of sensors is divided into two subarrays and the delay estimate is accomplished by cross-correlating the outputs of the two beamformers. Apparently, this method needs the direction of arrival (DOA) or TDOA as *a priori* information for subarray beamforming. Unfortunately, such information is hard to acquire if not available in reality. In [31], Griebel and Brandstein offered a consistency-based method where multiple sensors are partitioned into several pairs, and cross-correlation functions from different sensor pairs are fused together in the final cost function to obtain the time delay. This method requires the sensors to be paired in such a way that each correlation function should have a peak due to the same source in the same location. We have recently proposed a TDE algorithm based on the spatial interpolation technique, which will be detailed later. This method is shown to be a natural generalization of the cross-correlation method. It can improve the TDE performance as the number of sensors increases, without any *a priori* knowledge.

The second effort to deal with reverberation is to remodel the observation signals. In [15], a convolutive model is proposed to describe the TDE problem.

Different from the traditional way, this new model takes into account not only the direct path, but all the reflections as well, whereby the received signal of each sensor is modeled as the convolution of the source signal with the channel impulse response from the source to the sensor. The TDE problem then is rooted on identifying the channel impulse responses from the source to the sensors. Also in [15], an adaptive eigenvalue decomposition based algorithm is introduced to estimate two channel impulse responses (blindly), and thus the TDOA between the two channels. This method yields a robust solution to the TDE problem in a reverberant environment when the two channels do not share common zeros. This method is further enhanced by a multichannel technique presented in [32].

The objective of this chapter is to review some recent developments made in the TDE research. The focus is placed on their performance analysis and comparison in reverberant environments. In particular, this chapter reviews the generalized cross-correlation algorithm, the multichannel cross-correlation algorithm, and the blind-channel identification technique based algorithms along with their relations and improvements.

2. SIGNAL MODELS

Three models have been employed to describe an acoustic environment in the TDE problem: the ideal single-path propagation model, the multipath model, and the reverberant model.

2.1 IDEAL PROPAGATION MODEL

This model assumes that the signal acquired by each sensor is a delayed and attenuated version of the original source signal plus some additive noise. Suppose that we have an array consisting of $L+1$ receivers, the received signals can be expressed as:

$$\begin{bmatrix} x_0[n] \\ x_1[n] \\ x_2[n] \\ \vdots \\ x_L[n] \end{bmatrix} = \begin{bmatrix} \alpha_0 & 0 & 0 & \cdots & 0 \\ 0 & \alpha_1 & 0 & \cdots & 0 \\ 0 & 0 & \alpha_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & 0 & \alpha_L \end{bmatrix} \begin{bmatrix} s[n-t] \\ s[n-t-\tau] \\ s[n-t-f_2(\tau)] \\ \vdots \\ s[n-t-f_L(\tau)] \end{bmatrix} + \begin{bmatrix} w_0[n] \\ w_1[n] \\ w_2[n] \\ \vdots \\ w_L[n] \end{bmatrix}, \tag{8.1}$$

where α_l, $l = 0, 1, 2, \cdots, L$, are the attenuation factors due to propagation effects, t is the propagation time from the unknown source $s[n]$ to Sensor 0, $w_l[n]$ is an additive noise signal at the lth microphone, τ is the relative delay between Microphones 0 and 1, and $f_l(\tau)$ is the relative delay between Microphones 0 and l. Thefunction f_l depends not only on τ but also on the microphone array geometry. For example, in the far-field case (plane wave propagation), for a linear and equispaced array, we have:

$$f_l(\tau) = l\tau, \tag{8.2}$$

and for a linear but non-equispaced array, we have:

$$f_l(\tau) = \frac{\sum_{i=0}^{l-1} d_i}{d_0}\tau, \tag{8.3}$$

where d_i is the distance between Microphones i and $i+1$, $i = 0, 1, 2, \cdots, L-1$. In the near-field case, f_l depends also on the position of the source. Also note that $f_l(\tau)$ can be a *nonlinear* function of τ for a nonlinear array geometry, even in the far-filed case (e.g., 3 equilateral sensors). In general τ is not known, but the geometry of the array is known such that the mathematical formulation of $f_l(\tau)$ is well defined or given. It is further assumed that $w_l(n)$ is a zero-mean Gaussian random process that is uncorrelated with $s(n)$ and the noise signals at other sensors. For this model, the TDE problem is formulated to determine an estimate $\hat{\tau}$ of the true time delay τ using a finite set of observation samples.

2.2 MULTIPATH MODEL

The ideal propagation model takes into account the direct path signal only. In many situations, however, each sensor receives multiple delayed and attenuated replicas of the source signal due to reflections of the source wavefront from boundaries and objects in addition to the direct path signal. This so-called multipath effect has been intensively studied in the literature [13, 14, 33, 34]. In this case, the received signals are often described mathematically as:

$$x_l[n] = \sum_{m=1}^{M} \alpha_{lm} s[n - t - \tau_{lm}] + w_l[n], \quad l = 0, 1, \cdots, L, \tag{8.4}$$

where α_{lm} is the attenuation factor from the unknown source to the lth sensor via the mth path, t is the propagation time from the source to Sensor 0 via direct path, τ_{lm} is the relative delay between Sensor l and Sensor 0 for path m with $\tau_{01} = 0$, M is the number of different paths, and $w_l[n]$ is stationary Gaussian noise and assumed to be uncorrelated with both the source signal and the noise signals received by other sensors. This model is widely adopted in the oceanic propagation environments as illustrated in Fig. 8.1, where each sensor receives the direct path signal, as well as reflections from both the sea surface and the sea bottom [35, 36]. The primary interest of the TDE problem for this model is to measure τ_{l1}, $l = 1, \cdots, L$, which is the TDOA between Sensor l and Sensor 0 via direct path.

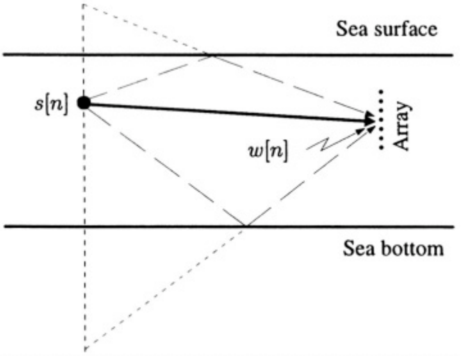


Figure 8.1 Illustration of the signal model in a multipath environment.

2.3 REVERBERANT MODEL

The multipath model is valid for some but not all environments [37]. In addition, if there are many different paths, i.e., M is large, it is difficult to estimate all τ_{lm}'s in (8.4). Recently, a more realistic convolutive model has been introduced to describe the TDE problem in a room environment where each sensor often receives a large number of echoes due to the reflections of objects and room boundaries such as walls, ceiling, and floor [15]. In addition, reflections can occur several times before a signal reaches the array, as shown in Fig. 8.2. In this model, the received signals are expressed as:

$$x_l[n] = h_l * s[n] + w_l[n], \tag{8.5}$$

where $*$ denotes convolution, h_l is the channel impulse response between the source and the lth sensor, and again $w_l[n]$ is a noise signal.

As seen, no time delay is explicitly expressed in (8.5), hence there is no plain solution to the TDE problem for the reverberation model, unless the channel impulse responses can be accurately (and blindly) identified, which is a very challenging problem.

3. GENERALIZED CROSS-CORRELATION METHOD

The generalized cross-correlation (GCC) algorithm is the most widely used approach for TDE, which is based on the ideal propagation model with two sensors, i.e., (8.1) with $L = 1$. In this framework, the delay estimate is obtained as

$$\hat{\tau}_{\text{GCC}} = \arg\max_n \hat{\Psi}_{\text{GCC}}[n], \tag{8.6}$$

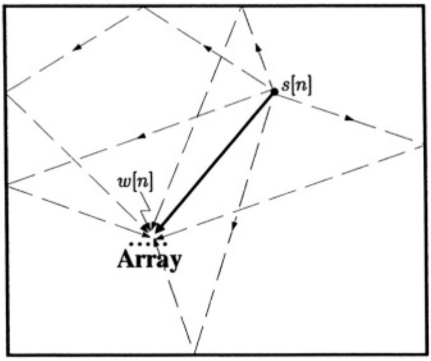


Figure 8.2 Illustration of the signal model in a reverberant environment.

where

$$\hat{\Psi}_{\text{GCC}}[n] = \sum_{k=0}^{N-1} \Phi[k] S_{x_0 x_1}[k] e^{\frac{j2\pi nk}{N}}$$

is the generalized cross-correlation function, $S_{x_0x_1}[k] = E\{X_0[k]X_1^*[k]\}$ is the cross spectrum, $E\{\cdot\}$ and $(\cdot)^*$ stand respectively for the mathematical expectation and the complex conjugate operator, $X_l[k]$ is the discrete Fourier transform (DFT) of $x_l[n]$, $\Phi[k]$ is a weighting function (sometimes called a *prefilter*), and N denotes the number of observation samples during the observation interval.

The weighting function $\Phi[k]$ plays an important role in controlling the TDE performance. It is chosen according to some criterion. Commonly used weighting functions include unit weighting (the classical cross-correlation method), the smoothed coherence transform (SCOT) [38], the Roth processor [39], the Echart filter, the phase transform (PHAT), the maximum likelihood (ML) processor[5], the Hassab-Boucher transform [16], etc. Some of these are optimal in the sense that the estimation variance can achieve the Cramèr-Rao lower bound (CRLB). Others are suboptimal but possess special properties, as for example the PHAT algorithm where $\Phi_{\text{PHAT}}[k] = 1/|S_{x_0x_1}[k]|$. Substituting $\Phi_{\text{PHAT}}[k]$ into (8.6) and neglecting noise effects, one can readily deduce that the weighted cross spectrum is free from the source signal and depends only on the channel responses. Consequently the PHAT algorithm performs more consistently than many other GCC members when the characteristics of the source signal change. This makes the PHAT algorithm superior in many applications.

GCC is a computationally efficient algorithm and is simple to implement. It performs well in the single-path propagation scenario when the signal-to-noise ratio (SNR) is high [5, 7, 16]. However, its performance degrades significantly when SNR drops below a certain threshold. This so-called threshold effect is also observed in a reverberant environment where the GCC method suffers sudden performance degradation when the reverberation time increases up to around 0.15 s [29]. In addition, most weighting functions in the GCC family are dependent on the signal and noise spectra and, in the absence of this prior information, the spectra can only be estimated. Therefore even though certain weighting functions are known to be optimal in theory, they can only be approximated in practice.

4. THE MULTICHANNEL CROSS-CORRELATION ALGORITHM

The fact that the GCC method does not perform well in reverberant environments motivated research to develop new algorithms. Two strategies were adopted. One is to blindly estimate the impulse responses from the source to various sensors, which will be discussed later. The other is to exploit the redundancy among different sensor signals in an array for robustness against noise and reverberation and will be developed here.

4.1 SPATIAL PREDICTION TECHNIQUE

As seen from the ideal propagation model given in (8.1), the signal of one sensor is not completely independent from the other sensor signals. The spatial prediction technique captures the dependence among signals and makes a prediction of a data sample of one sensor using samples from L other sensors. This concept was presented in [40] for the simple case in which the spatial prediction is made equivalent to the classical linear prediction. In this section, we generalize this idea in a way that the geometry of the array is taken into account as well as the relative delay among the elements. As a result, the spatial correlation matrix has a much more general form.

4.1.1 Linear Forward Spatial Prediction. We would like to align successive time samples of Microphone 0 signal with spatial samples from the L other microphone signals. It is clear that $x_0[n - f_L(\tau)]$ is inphase with the signals $x_l[n - f_L(\tau) + f_l(\tau)]$, $l = 1, 2, \cdots, L$. From these observations, we define the following forward spatial prediction error signal:

$$e_0[n - f_L(m)] = x_0[n - f_L(m)] - \mathbf{x}_{1:L}^T[n - f_L(m)]\mathbf{a}_m, \tag{8.7}$$

where m is a guessed relative delay for τ, $(\cdot)^T$ denotes transpose of a vector or a matrix,

$$\mathbf{x}_{1:L}[n - f_L(m)] = \\ \begin{bmatrix} x_1[n - f_L(m) + f_1(m)] & x_2[n - f_L(m) + f_2(m)] & \cdots & x_L[n] \end{bmatrix}^T,$$

and

$$\mathbf{a}_m = \begin{bmatrix} a_{m,1} & a_{m,2} & \cdots & a_{m,L} \end{bmatrix}^T$$

is the linear forward spatial predictor. Consider the criterion

$$J_{m,0} = E\{e_0^2[n - f_L(m)]\}. \tag{8.8}$$

Minimization of (8.8) leads to the equation:

$$\mathbf{R}_{m,1:L}\mathbf{a}_m = \mathbf{r}_{m,1:L}, \tag{8.9}$$

where

$$\begin{aligned}\mathbf{R}_{m,1:L} &= E\{\mathbf{x}_{1:L}[n - f_L(m)]\mathbf{x}_{1:L}^T[n - f_L(m)]\} \\ &= \begin{bmatrix} E\{x_1^2[n]\} & R_{12}(m) & \cdots & R_{1L}(m) \\ R_{21}(m) & E\{x_2^2[n]\} & \cdots & R_{2L}(m) \\ \vdots & \vdots & \ddots & \vdots \\ R_{L1}(m) & R_{L2}(m) & \cdots & E\{x_L^2[n]\} \end{bmatrix}\end{aligned}$$

is the spatial correlation matrix with

$$R_{lk}(m) = E\{x_l[n - f_k(m)]x_k[n - f_l(m)]\},$$

and

$$\begin{aligned}\mathbf{r}_{m,1:L} &= E\{\mathbf{x}_{1:L}[n - f_L(m)]x_0[n - f_L(m)]\} \\ &= \begin{bmatrix} E\{x_1[n - f_L(m) + f_1(m)]x_0[n - f_L(m)]\} \\ E\{x_2[n - f_L(m) + f_2(m)]x_0[n - f_L(m)]\} \\ \vdots \\ E\{x_L[n]x_0[n - f_L(m)]\} \end{bmatrix} \\ &= \begin{bmatrix} E\{x_1[n]x_0[n - f_1(m)]\} \\ E\{x_2[n]x_0[n - f_2(m)]\} \\ \vdots \\ E\{x_L[n]x_0[n - f_L(m)]\} \end{bmatrix}\end{aligned}$$

is the spatial correlation vector. Note that the spatial correlation matrix is not Toeplitz in general, except in some particular cases.

For $m = \tau$ and for the noise-free case (where $w_l[n] = 0, l = 0, 1, 2, \cdots, L$), it can easily be checked that with the ideal propagation model, the rank of matrix $\mathbf{R}_{\tau,1:L}$ is equal to 1. This means that the samples $x_0[n - \tau]$ can be perfectly predicted from any one of the other microphone samples. However, the noise is never absent in practice and is in general isotropic. The noise energy at different microphones is added to the main diagonal of the correlation matrix $\mathbf{R}_{\tau,1:L}$, which will regularize it and make it positive definite (which we suppose in the rest of this chapter). A unique solution to (8.9) is then guaranteed for any number of microphones. This solution is optimal from a Wiener theory point of view.

4.1.2 Linear Backward Spatial Prediction. We still assume the ideal propagation model but this time we consider Microphone L and we would like to align successive time samples of this microphone signal with spatial samples from the L other microphone signals. It is clear that $x_L[n]$ is inphase with the signals $x_l[n - f_L(\tau) + f_l(\tau)]$, $l = 0, 1, \cdots, L-1$. From these observations, we define the following backward spatial prediction error signal:

$$e_L[n - f_L(m)] = x_L[n] - \mathbf{x}_{0:L-1}^T[n - f_L(m)]\mathbf{b}_m, \tag{8.10}$$

where

$$\begin{aligned}\mathbf{x}_{0:L-1}[n - f_L(m)] &= \\ &\begin{bmatrix} x_0[n - f_L(m) + f_0(m)] & x_1[n - f_L(m) + f_1(m)] & \cdots \\ & x_{L-1}[n - f_L(m) + f_{L-1}(m)] \end{bmatrix}^T\end{aligned}$$

and

$$\mathbf{b}_m = \begin{bmatrix} b_{m,0} & b_{m,1} & \cdots & b_{m,L-1} \end{bmatrix}^T$$

is the linear backward spatial predictor. Minimization of the criterion

$$J_{m,L} = E\{e_L^2[n - f_L(m)]\} \tag{8.11}$$

leads to the equation:

$$\mathbf{R}_{m,0:L-1}\mathbf{b}_m = \mathbf{r}_{m,0:L-1}, \tag{8.12}$$

where

$$\mathbf{R}_{m,0:L-1} = E\{\mathbf{x}_{0:L-1}[n - f_L(m)]\mathbf{x}_{0:L-1}^T[n - f_L(m)]\}$$

and

$$\mathbf{r}_{m,0:L-1} = E\{\mathbf{x}_{0:L-1}[n - f_L(m)]x_L[n]\}.$$

4.1.3 Linear Spatial Interpolation. The ideas presented for spatial prediction can easily be extended to spatial interpolation, where we consider any microphone element l, $l = 0, 1, 2, \cdots, L$. The spatial interpolation error signal is defined as

$$e_l[n - f_L(m)] = -\mathbf{x}_{0:L}^T[n - f_L(m)]\mathbf{c}_{m,l}, \tag{8.13}$$

where

$$\begin{aligned}\mathbf{x}_{0:L}[n - f_L(m)] &= \\ &\begin{bmatrix} x_0[n - f_L(m) + f_0(m)] & x_1[n - f_L(m) + f_1(m)] & \cdots & x_L[n] \end{bmatrix}^T\end{aligned}$$

and

$$\mathbf{c}_{m,l} = \begin{bmatrix} c_{m,l,0} & c_{m,l,1} & \cdots & c_{m,l,L} \end{bmatrix}^T$$

is the spatial interpolator with $c_{m,l,l} = -1$. The criterion associated with (8.13) is:

$$J_{m,l} = E\{e_l^2[n - f_L(m)]\}. \tag{8.14}$$

The rest flows immediately from the previous sections on spatial prediction.

4.2 TIME DELAY ESTIMATION USING SPATIAL PREDICTION

The spatial prediction, and more generally the spatial interpolation technique can be applied to the problem of TDE. Here, we consider the linear forward prediction only. The idea can be easily generalized to the linear backward prediction and spatial interpolation.

Let $J_{m,0;\min}$ denote the minimum mean-squared error, for the value m, defined by

$$J_{m,0;\min} = E\{e_{0;\min}^2[n - f_L(m)]\}. \tag{8.15}$$

If we replace $\mathbf{a}_m$ by $\mathbf{R}_{m,1:L}^{-1}\mathbf{r}_{m,1:L}$ in (8.7), we get:

$$\begin{aligned} e_{0;\min}[n - f_L(m)] &= \\ & x_0[n - f_L(m)] - \mathbf{x}_{1:L}^T[n - f_L(m)]\mathbf{R}_{m,1:L}^{-1}\mathbf{r}_{m,1:L}. \end{aligned} \tag{8.16}$$

We deduce that:

$$J_{m,0;\min} = E\{x_0^2[n - f_L(m)]\} - \mathbf{r}_{m,1:L}^T\mathbf{R}_{m,1:L}^{-1}\mathbf{r}_{m,1:L}. \tag{8.17}$$

The value of m that gives the minimum $J_{m,0;\min}$, for different m, corresponds to the time delay between Microphones 0 and 1. Mathematically, the solution to the TDE problem is then given by

$$\hat{\tau} = \arg\min_m J_{m,0;\min}, \tag{8.18}$$

where $\hat{\tau}$ is an estimate of τ.

Particular case: Two microphones ($L = 1$). In this case, the solution is:

$$\begin{aligned} \hat{\tau} &= \arg\min_m \left\{ E\{x_0^2[n-m]\} \left[1 - \frac{E^2\{x_0[n-m]x_1[n]\}}{E\{x_0^2[n-m]\}E\{x_1^2[n]\}} \right] \right\} \\ &= \arg\min_m \left\{ E\{x_0^2[n-m]\} \left[1 - \rho_{m,01}^2 \right] \right\} \\ &= \arg\min_m \left\{ 1 - \rho_{m,01}^2 \right\} \\ &= \arg\max_m \left(\rho_{m,01}^2 \right), \end{aligned} \tag{8.19}$$

where $\rho_{m,01}$ ($\rho_{m,01}^2 \leq 1$) is the cross-correlation coefficient between $x_0[n-m]$ and $x_1[n]$. When the cross-correlation coefficient is close to 1, this means that the two signals that we compare are highly correlated which happens when the signals are inphase, i.e., $m \approx \tau$, and this implies that $J_{\tau,0;\min} \approx 0$. This approach is similar to the GCC method. Note that in the general case with any number of microphones, the proposed approach can be seen as a cross-correlation method. However, we take advantage of the knowledge of the microphone array to estimate only one time delay (instead of estimating multiple time delays independently) in an optimal way in a least mean square sense.

4.3 OTHER INFORMATION FROM THE SPATIAL CORRELATION MATRIX

Consider the $L+1$ microphone signals x_l, $l = 0, 1, \cdots, L$. The corresponding spatial correlation matrix is

$$\begin{aligned} \mathbf{R}_m &= \mathbf{R}_{m,0:L} \\ &= E\{\mathbf{x}_{0:L}[n - f_L(m)]\mathbf{x}_{0:L}^T[n - f_L(m)]\}, \end{aligned} \tag{8.20}$$

which can be factored as:

$$\mathbf{R}_m = \mathbf{D}\widetilde{\mathbf{R}}_m\mathbf{D}, \tag{8.21}$$

where

$$\mathbf{D} = \begin{bmatrix} \sqrt{E\{x_0^2[n]\}} & 0 & \cdots & 0 \\ 0 & \sqrt{E\{x_1^2[n]\}} & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \sqrt{E\{x_L^2[n]\}} \end{bmatrix} \tag{8.22}$$

is a diagonal matrix,

$$\widetilde{\mathbf{R}}_m = \begin{bmatrix} 1 & \rho_{m,01} & \cdots & \rho_{m,0L} \\ \rho_{m,01} & 1 & \cdots & \rho_{m,1L} \\ \vdots & \ddots & \ddots & \vdots \\ \rho_{m,0L} & \cdots & \rho_{m,L-1L} & 1 \end{bmatrix} \tag{8.23}$$

is a symmetric matrix, and

$$\rho_{m,kl} = \frac{E\{x_k[n - f_l(m)]x_l[n - f_k(m)]\}}{\sqrt{E\{x_k^2[n]\}E\{x_l^2[n]\}}}, \quad k, l = 0, 1, \cdots, L, \tag{8.24}$$

is the cross-correlation coefficient between $x_k[n - f_l(m)]$ and $x_l[n - f_k(m)]$.

We now give two propositions that will be useful for TDE.

Proposition 1. A spatial cross-correlation coefficient matrix $\widetilde{\mathbf{R}}_m$ satisfies

$$0 < \det\left(\widetilde{\mathbf{R}}_m\right) \leq 1, \tag{8.25}$$

where "det" stands for *determinant.*

Proof. Since $\mathbf{R}_m$ is symmetric and is supposed to be positive definite, it is clear that $\det\left(\mathbf{R}_m\right) > 0$ which implies that $\det(\widetilde{\mathbf{R}}_m) > 0$. To show that $\det(\widetilde{\mathbf{R}}_m) \leq 1$, we can use the *Cholesky factorization* [41]. Since $\widetilde{\mathbf{R}}_m$ is symmetric and positive definite, there exists a unique lower triangular matrix $\mathbf{Q}_m$ with positive diagonal entries such that $\widetilde{\mathbf{R}}_m = \mathbf{Q}_m\mathbf{Q}_m^T$, where

$$\mathbf{Q}_m = \begin{bmatrix} q_{m,00} & 0 & \cdots & 0 \\ q_{m,10} & q_{m,11} & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ q_{m,L0} & \cdots & q_{m,LL-1} & q_{m,LL} \end{bmatrix}. \tag{8.26}$$

It can be shown that the elements of the main diagonal of matrix $\mathbf{Q}_m$ can be computed as follows:

$$q_{m,ll} = \sqrt{1 - \sum_{k=0}^{l-1} q_{m,lk}^2}, \quad l = 0, 1, \cdots, L. \tag{8.27}$$

It follows immediately from (8.27) that $0 < q_{m,ll} \leq 1$, $\forall l$. Furthermore, since $\mathbf{Q}_m$ is a triangular matrix, we have:

$$\det\left(\widetilde{\mathbf{R}}_m\right) = \prod_{l=0}^{L} q_{m,ll}^2 \leq 1,$$

which completes the proof.

Another way to show this proposition is by induction, i.e.,

$$\det\left(\widetilde{\mathbf{R}}_m\right) = \det\left(\widetilde{\mathbf{R}}_{m,0:L}\right) \leq \det\left(\widetilde{\mathbf{R}}_{m,1:L}\right) \leq \cdots \leq 1. \tag{8.28}$$

Proposition 2. The determinant of a cross-correlation coefficient matrix is bounded by

$$\det\left(\widetilde{\mathbf{R}}_m\right) \leq \frac{J_{m,0;\min}}{E\{x_0^2[n]\}} \leq 1. \tag{8.29}$$

Proof. First define

$$\underline{\mathbf{a}}_m = \left[\begin{array}{cc} a_{m,0} & \mathbf{a}_m^T \end{array} \right]^T. \tag{8.30}$$

Then, for $a_{m,0} = -1$, the forward prediction error signal defined in (8.7) can be rewritten as

$$e_0[n - f_L(m)] = -\mathbf{x}_{0:L}^T \underline{\mathbf{a}}_m. \tag{8.31}$$

Continuing, the criterion shown in (8.8) can be expressed as

$$J_{m,0} = E\{e_0^2[n - f_L(m)]\} + \lambda(\boldsymbol{\delta}^T \underline{\mathbf{a}}_m + 1), \tag{8.32}$$

where $\boldsymbol{\delta} = [1\ 0\ \cdots\ 0]^T$, and λ is a Lagrange multiplier introduced to force $a_{m,0}$ to have value -1. It is then easily shown that:

$$J_{m,0;\min} = \frac{1}{\boldsymbol{\delta}^T \mathbf{R}_m^{-1} \boldsymbol{\delta}}. \tag{8.33}$$

In this case, with (8.21), (8.33) becomes:

$$\begin{aligned} J_{m,0;\min} &= \frac{E\{x_0^2[n]\}}{\boldsymbol{\delta}^T \widetilde{\mathbf{R}}_m^{-1} \boldsymbol{\delta}} \\ &= E\{x_0^2[n]\} \frac{\det\left(\widetilde{\mathbf{R}}_m\right)}{\det\left(\widetilde{\mathbf{R}}_{m,1:L}\right)}. \end{aligned} \tag{8.34}$$

Using (8.28), it is clear that Proposition 2 is verified.

In the general case, for any interpolator, we have:

$$\det\left(\widetilde{\mathbf{R}}_m\right) \leq \frac{J_{m,l;\min}}{E\{x_l^2[n]\}} \leq 1,\ l = 0, 1, \cdots, L. \tag{8.35}$$

As seen, the determinant of the spatial correlation matrix is related to the minimum mean-squared error and to the power of the signals. Let's take the two-channel case. It is obvious that the cross-correlation coefficient between the two signals x_0 and x_1 is linked to the determinant of the corresponding spatial correlation matrix:

$$\rho_{m,01}^2 = 1 - \det\left(\widetilde{\mathbf{R}}_{m,0:1}\right). \tag{8.36}$$

By analogy to the cross-correlation coefficient definition between two random signals, we define the multichannel correlation coefficient among the signals x_l, $l = 0, 1, \cdots, L$, as:

$$\rho_{m,0:L}^2 = 1 - \det\left(\widetilde{\mathbf{R}}_{m,0:L}\right). \tag{8.37}$$

From proposition 2, we give a new bound for $\rho^2_{m,0:L}$:

$$1 - \frac{J_{m,0;\min}}{E\{x_0^2[n]\}} \leq \rho^2_{m,0:L} \leq 1. \tag{8.38}$$

Basically, the coefficient $\rho_{m,0:L}$ will measure the amount of correlation among all the channels. This coefficient has some interesting properties. For example, if one of the signals, say x_0, is completely decorrelated from the others because the microphone is defective, or it picks up only noise, or the signal is saturated, this signal will not affect $\rho_{m,0:L}$ since $\rho_{m,0l} = 0$, $\forall l$. In this case:

$$\rho^2_{m,0:L} = \rho^2_{m,1:L}. \tag{8.39}$$

In other words, the measure "drops" the signals that have no correlation with the others. This makes sense from a correlation point of view, since we want to measure the degree of correlation only from the channels that have something in common. In the extreme cases where all the signals are uncorrelated, we have $\rho^2_{m,0:L} = 0$, and where any two signals (or more) are perfectly correlated, we have $\rho^2_{m,0:L} = 1$.

Obviously, the multichannel coefficient $\rho^2_{m,0:L}$ can be used for time delay estimation in the following way:

$$\begin{aligned} \hat{\tau} &= \arg\max_m \left(\rho^2_{m,0:L}\right) \\ &= \arg\min_m \left[\det\left(\widetilde{\mathbf{R}}_{m,0:L}\right)\right]. \end{aligned} \tag{8.40}$$

This method can be seen as a multichannel correlation approach to the estimation of time delay and it is clear that (8.40) is equivalent to (8.18).

5. ADAPTIVE EIGENVALUE DECOMPOSITION ALGORITHM

Although the TDE performance can be improved by empolying multiple sensors, the multichannel cross-correlation method is still based on the ideal propagation model which takes into account merely the direct paths. Starting from here, we intend to approach this problem from a different direction, considering the more realistic reverberant model and using the blind multichannel identification technique. Again we will first focus on an array with only two channels and develop the adaptive eigenvalue decomposition algorithm. Then we proceed to generalize the idea to multichannel cases.

With the reverberant model given in (8.5) with two sensors, if the noise term is neglected, one can readily derive the following relation:

$$x_0[n] * h_1 = s[n] * h_0 * h_1 = x_1[n] * h_0. \tag{8.41}$$

At time instant n, this relation can be rewritten in a vector-matrix form as [15]:

$$\mathbf{x}^T[n]\mathbf{u} = \mathbf{x}_0^T[n]\mathbf{h}_1 - \mathbf{x}_1^T[n]\mathbf{h}_0 = 0, \tag{8.42}$$

where

$$\begin{aligned}
\mathbf{x}_l[n] &= \left[\begin{array}{cccc} x_l[n] & x_l[n-1] & \cdots & x_l[n-M+1] \end{array}\right]^T, \\
\mathbf{h}_l &= \left[\begin{array}{cccc} h_{l,0} & h_{l,1} & \cdots & h_{l,M-1} \end{array}\right]^T, \\
\mathbf{x}[n] &= \left[\begin{array}{cc} \mathbf{x}_0^T[n] & \mathbf{x}_1^T[n] \end{array}\right]^T, \\
\mathbf{u} &= \left[\begin{array}{cc} \mathbf{h}_0^T & -\mathbf{h}_1^T \end{array}\right]^T,
\end{aligned}$$

$l = 0, 1$, and M is the length of the impulse responses. Left multiplying (8.42) by $\mathbf{x}[n]$ and taking expectation yields

$$\mathbf{R}[n]\mathbf{u} = 0, \tag{8.43}$$

where $\mathbf{R}[n] = E\{\mathbf{x}[n]\mathbf{x}^T[n]\}$ is the covariance matrix of the microphone signals. This implies that vector $\mathbf{u}$ which consists of two impulse responses is in the null space of $\mathbf{R}[n]$. More specifically, $\mathbf{u}$ is the eigenvector of $\mathbf{R}[n]$ corresponding to the eigenvalue 0. This remarkable observation forms the basis of the eigenvector based TDE algorithm [15, 42].

Before going further, we need to know whether (8.43) has a unique solution (up to arbitrary constant) other than the trivial solution. Indeed, it was shown in [43] that it is the case if the following conditions hold:

- The polynomials $H_0(z)$ and $H_1(z)$ are co-prime, or equivalently, they do not share any common zeros, where $H_0(z)$ and $H_1(z)$ are the z-transforms of $\mathbf{h}_0$, $\mathbf{h}_1$ respectively.

- The autocorrelation matrix of the source signal $s(n)$ is full rank.

When an independent white noise is present on each of the two microphones, it will regularize the covariance matrix; as a consequence, $\mathbf{R}[n]$ does not have a zero eigenvalue anymore. In such a case, the TDE problem can be formulated as to estimate $\mathbf{u}$ by minimizing $\mathbf{u}^T\mathbf{R}[n]\mathbf{u}$ subject to $||\mathbf{u}|| = 1$. This is equivalent to finding the normalized eigenvector associated with the lowest eigenvalue of $\mathbf{R}[n]$ [15].

With the background that we have developed, we are now in a position to treat the subject of efficiently estimating the eigenvector corresponding to the smallest eigenvalue of $\mathbf{R}[n]$. In principle, any eigenvalue decomposition algorithm can be used to solve the problem. Here we choose the constrained LMS adaptive algorithm [15] for its simplicity, efficiency, and ability to compensate for slow environmental changes. With such a method, an estimate of $\mathbf{u}$ is updated iteratively through:

$$\hat{\mathbf{u}}[n+1] = \frac{\hat{\mathbf{u}}[n] - \mu e[n]\mathbf{x}[n]}{||\hat{\mathbf{u}}[n] - \mu e[n]\mathbf{x}[n]||}, \tag{8.44}$$

and

$$e[n] = \hat{\mathbf{u}}^T[n]\mathbf{x}[n], \tag{8.45}$$

with the constraint that $||\hat{\mathbf{u}}(n)|| = 1$, where μ, the adaptation step, is a positive constant.

In practice, (8.44) may not produce an accurate estimation of the impulse responses because of the nonstationarity of speech, the background noise, and the unknown length of the impulse responses. However, it yields a solution that is accurate enough for the purpose of TDE since in such an application only two direct paths are of interest.

6. ADAPTIVE MULTICHANNEL TIME DELAY ESTIMATION

In the adaptive eigenvalue decomposition (AED) algorithm, the delay estimate is obtained by blindly identifying two channel impulse responses. It requires that the two channels do not share any common zeros, which is usually true for systems with short impulse responses. In many application scenarios such as room acoustic environments, however, the channel impulse response from the source to the microphone sensor could be very long. As a result, the likelihood for two impulse responses not sharing common zeros tends to be low and the AED algorithm often fails when a zero is shared between two channels or some zeros of the two channels are close. One way to overcome this problem is to employ more channels in the system, since it would be less likely for all channels to share a common zero when the number of sensors is large. This idea leads to an adaptive multichannel time delay estimation approach based on a blind channel identification technique [32].

6.1 PRINCIPLE

Generalizing the approach of Section 5 to more than two channels, we have in the noiseless case:

$$x_i[n] * h_j = s(n) * h_i * h_j = x_j[n] * h_i, \quad i,j = 0,1,\cdots,L, \tag{8.46}$$

and the vector-matrix form of cross relation between the ith and jth channel outputs is

$$\mathbf{x}_i^T[n]\mathbf{h}_j = \mathbf{x}_j^T[n]\mathbf{h}_i, \quad i,j = 0,1,2,\cdots,L,\ i \neq j. \tag{8.47}$$

When noise is present or the channel impulse responses are improperly modeled, the left and right hand sides of (8.47) are generally not equal and the inequality can be used to define an error signal at time $n+1$ as follows:

$$e_{ij}[n+1] = \frac{\mathbf{x}_i^T[n+1]\hat{\mathbf{h}}_j[n] - \mathbf{x}_j^T[n+1]\hat{\mathbf{h}}_i[n]}{\left\|\hat{\mathbf{h}}[n]\right\|}, \quad i,j = 0,1,\cdots,L, \quad (8.48)$$

where

$$\hat{\mathbf{h}}_i[n] = \begin{bmatrix} \hat{h}_{i,0}[n] & \hat{h}_{i,1}[n] & \cdots & \hat{h}_{i,M-1}[n] \end{bmatrix}^T$$

is the modeling filter for the ith channel at time n and

$$\hat{\mathbf{h}}[n] = \begin{bmatrix} \hat{\mathbf{h}}_0^T[n] & \hat{\mathbf{h}}_1^T[n] & \cdots & \hat{\mathbf{h}}_L^T[n] \end{bmatrix}^T.$$

The modeling filter is normalized in order to avoid a trivial solution whose elements are all zero. Based on the error signal defined here, a cost function at time $n+1$ is given by

$$J[n+1] = \sum_{i=0}^{L-1} \sum_{j=i+1}^{L} e_{ij}^2[n+1]. \quad (8.49)$$

The TDE problem is then to obtain the estimate of $\mathbf{h}_i/\|\mathbf{h}\|$ $(i = 0,1,\cdots,L)$ that minimizes this cost function.

6.2 TIME-DOMAIN MULTICHANNEL LMS APPROACH

A straightforward approach to estimating channel impulse responses from the cost function defined in (8.49) is through the multichannel LMS (MCLMS) algorithm [44], which updates $\hat{\mathbf{h}}$ through:

$$\hat{\mathbf{h}}[n+1] = \hat{\mathbf{h}}[n] - \mu\nabla J[n+1], \quad (8.50)$$

where μ is a small positive step size. As shown in [44], the gradient of $J[n+1]$ is determined as:

$$\nabla J[n+1] = \frac{\partial J[n+1]}{\partial\hat{\mathbf{h}}[n]} = \frac{2\left[\tilde{\mathbf{R}}[n+1]\hat{\mathbf{h}}[n] - J[n+1]\hat{\mathbf{h}}[n]\right]}{\|\hat{\mathbf{h}}[n]\|^2}, \quad (8.51)$$

where

$$\tilde{\mathbf{R}}[n+1] = \begin{bmatrix} \sum_{i\neq 0}\tilde{\mathbf{R}}_{x_ix_i}[n+1] & -\tilde{\mathbf{R}}_{x_1x_0}[n+1] & \cdots & -\tilde{\mathbf{R}}_{x_Lx_0}[n+1] \\ -\tilde{\mathbf{R}}_{x_0x_1}[n+1] & \sum_{i\neq 1}\tilde{\mathbf{R}}_{x_ix_i}[n+1] & \cdots & -\tilde{\mathbf{R}}_{x_Lx_1}[n+1] \\ \vdots & \vdots & \ddots & \vdots \\ -\tilde{\mathbf{R}}_{x_0x_L}[n+1] & -\tilde{\mathbf{R}}_{x_1x_L}[n+1] & \cdots & \sum_{i\neq L}\tilde{\mathbf{R}}_{x_ix_i}[n+1] \end{bmatrix},$$

and

$$\tilde{\mathbf{R}}_{x_i x_j}[n+1] = \mathbf{x}_i[n+1]\mathbf{x}_j^T[n+1], \quad i,j = 0,1,\cdots,L.$$

If the modeling filter is normalized after each update, then a simplified algorithm is obtained:

$$\hat{\mathbf{h}}[n+1] = \frac{\hat{\mathbf{h}}[n] - 2\mu\left[\tilde{\mathbf{R}}[n+1]\hat{\mathbf{h}}[n] - J[n+1]\hat{\mathbf{h}}[n]\right]}{\left\|\hat{\mathbf{h}}[n] - 2\mu\left[\tilde{\mathbf{R}}[n+1]\hat{\mathbf{h}}[n] - J[n+1]\hat{\mathbf{h}}[n]\right]\right\|}. \tag{8.52}$$

It was shown theoretically and demonstrated empirically in [44] that the MCLMS algorithm converges in mean to the desired impulse responses.

6.3 FREQUENCY-DOMAIN ADAPTIVE ALGORITHMS

The time-domain MCLMS algorithm is simple to implement and it converges steadily, but its convergence rate is slow. There are many ways to accelerate the convergence rate. One is to implement the adaptive algorithm in the frequency-domain, which takes advantage of the the fast Fourier transform (FFT) to make the adaptive process more efficient [32].

To begin, we define an intermediate signal $y_{ij} \triangleq x_i * \hat{h}_j$, the convolution result of the ith channel output x_i and the jth model filter $\hat{h}_j$. In vector form, a block of such a signal can be expressed in the frequency domain as

$$\underline{y}_{ij}[m+1] = \mathcal{W}^{01}_{M\times 2M}\mathcal{D}_{x_i}[m+1]\mathcal{W}^{10}_{2M\times M}\underline{\hat{h}}_j[m], \tag{8.53}$$

where

$$\begin{aligned}
\mathcal{W}^{01}_{M\times 2M} &= \mathbf{F}_{M\times M}\left[\begin{array}{cc} \mathbf{0}_{M\times M} & \mathbf{I}_{M\times M} \end{array}\right]\mathbf{F}^{-1}_{2M\times 2M},\\
\mathcal{D}_{x_i}[m+1] &= \mathrm{diag}\left\{\mathbf{F}_{2M\times 2M}\cdot \mathbf{x}_i[m+1]_{2M\times 1}\right\},\\
\mathcal{W}^{10}_{2M\times M} &= \mathbf{F}_{2M\times 2M}\left[\begin{array}{cc} \mathbf{I}_{M\times M} & \mathbf{0}_{M\times M} \end{array}\right]^T\mathbf{F}^{-1}_{M\times M},\\
\underline{\hat{h}}_j[m] &= \mathbf{F}_{M\times M}\hat{\mathbf{h}}_j[m],\\
\mathbf{x}_i[m+1]_{2M\times 1} &= \left[\begin{array}{cccc} x_i[mM] & x_i[mM+1] & \cdots & x_i[mM+2M-1] \end{array}\right]^T,
\end{aligned}$$

$\mathbf{F}_{M\times M}$ and $\mathbf{F}^{-1}_{M\times M}$ are respectively the Fourier and inverse Fourier matrices of size $M \times M$, and m is the block time index. Then a block of the error signal based on the cross relation between the ith and the jth channel in the frequency domain is determined as:

$$\begin{aligned}
\underline{e}_{ij}[m+1] &= \underline{y}_{ij}[m+1] - \underline{y}_{ji}[m+1]\\
&= \mathcal{W}^{01}_{M\times 2M}\Big[\mathcal{D}_{x_i}[m+1]\mathcal{W}^{10}_{2M\times M}\underline{\hat{h}}_j[m] -\\
&\qquad \mathcal{D}_{x_j}[m+1]\mathcal{W}^{10}_{2M\times M}\underline{\hat{h}}_i[m]\Big].
\end{aligned} \tag{8.54}$$

Continuing, we can construct a (frequency-domain) cost function at the $(m+1)$th time block index as follows:

$$J_{\mathrm{f}}[m+1] = \sum_{i=0}^{L-1} \sum_{j=i+1}^{L} \underline{e}_{ij}^{H}[m+1]\underline{e}_{ij}[m+1]. \tag{8.55}$$

Therefore, by minimizing $J_{\mathrm{f}}[m+1]$, the modeling filter can be updated in the frequency domain as:

$$\hat{\underline{h}}_k[m+1] = \hat{\underline{h}}_k[m] - \mu_{\mathrm{f}} \frac{\partial J_{\mathrm{f}}[m+1]}{\partial \hat{\underline{h}}_k^{*}[m]}, \quad k = 0, 1, \cdots, L, \tag{8.56}$$

where μ_{f} is a small positive step size. It can be shown that [23]

$$\frac{\partial J_{\mathrm{f}}[m+1]}{\partial \hat{\underline{h}}_k^{*}[m]} = \sum_{i=0}^{L} \left[\mathcal{W}_{M\times 2M}^{01} \mathcal{D}_{x_i}[m+1] \mathcal{W}_{2M\times M}^{10}\right]^{H} \underline{e}_{ik}[m+1]. \tag{8.57}$$

Substituting (8.57) into (8.56) yields the multichannel frequency-domain LMS (MCFLMS) algorithm:

$$\begin{aligned} &\hat{\underline{h}}_k[m+1] = \\ &\qquad \hat{\underline{h}}_k[m] - \mu_{\mathrm{f}} \mathcal{W}_{M\times 2M}^{10} \sum_{i=0}^{L} \mathcal{D}_{x_i}^{*}[m+1] \mathcal{W}_{2M\times M}^{01} \underline{e}_{ik}[m+1], \end{aligned} \tag{8.58}$$

where

$$\begin{aligned} \mathcal{W}_{M\times 2M}^{10} &= \mathbf{F}_{M\times M} \begin{bmatrix} \mathbf{I}_{M\times M} & \mathbf{0}_{M\times M} \end{bmatrix} \mathbf{F}_{2M\times 2M}^{-1}, \\ \mathcal{W}_{2M\times M}^{01} &= \mathbf{F}_{2M\times 2M} \begin{bmatrix} \mathbf{0}_{M\times M} & \mathbf{I}_{M\times M} \end{bmatrix}^{T} \mathbf{F}_{M\times M}^{-1}. \end{aligned}$$

A constraint ensuring that the adaptive algorithm would not converge to a trivial solution with all zero elements can be applied in either the frequency or the time domain. Since in the application of time delay estimation the delay will be estimated in the time domain and the time-domain modeling filter coefficients have to be computed anyway, we will enforce the constraint in the time domain for convenience.

The MCFLMS is computationally more efficient than a multichannel time-domain block LMS algorithm. However, the MCFLMS and its time-domain counterpart are equivalent in performance. The convergence of the MCFLMS algorithm is still slow because of nonuniform convergence rates of the filter coefficients and cross-coupling between them. To accelerate convergence, we will use Newton's method to develop a normalized MCFLMS (NMCFLMS) method.

By using Newton's method, the coefficients of the model filter can be updated according to:

$$\begin{aligned}\hat{\underline{h}}_k[m+1] &= \\ &\hat{\underline{h}}_k[m] - \mu_{\mathrm{f}} E^{-1}\left\{\frac{\partial}{\partial \hat{\underline{h}}_k^T[m]}\left[\frac{\partial J_{\mathrm{f}}[m+1]}{\partial \hat{\underline{h}}_k^*[m]}\right]\right\}\frac{\partial J_{\mathrm{f}}[m+1]}{\partial \hat{\underline{h}}_k^*[m]},\end{aligned} \tag{8.59}$$

where the Hessian matrix can be evaluated as

$$\begin{aligned}E\left\{\frac{\partial}{\partial \hat{\underline{h}}_k^T[m]}\left[\frac{\partial J_{\mathrm{f}}[m+1]}{\partial \hat{\underline{h}}_k^*[m]}\right]\right\} = \mathcal{W}_{M\times 2M}^{10}\cdot \\ \sum_{i=0,i\neq k}^{L}\left[\mathcal{D}_{x_i}^*[m+1]\mathcal{W}_{2M\times 2M}^{01}\mathcal{D}_{x_i}[m+1]\right]\mathcal{W}_{2M\times M}^{10},\end{aligned} \tag{8.60}$$

and

$$\begin{aligned}\mathcal{W}_{2M\times 2M}^{01} &\triangleq \mathcal{W}_{2M\times M}^{01}\mathcal{W}_{M\times 2M}^{01} \\ &= \mathbf{F}_{2M\times 2M}\begin{bmatrix}\mathbf{0}_{M\times M} & \mathbf{0}_{M\times M} \\ \mathbf{0}_{M\times M} & \mathbf{I}_{M\times M}\end{bmatrix}\mathbf{F}_{2M\times 2M}^{-1}.\end{aligned}$$

As shown in [23], when M is large, $2\mathcal{W}_{2M\times 2M}^{01}$ can be well approximated by the identity matrix

$$2\mathcal{W}_{2M\times 2M}^{01} \approx \mathbf{I}_{2M\times 2M}. \tag{8.61}$$

Thus, (8.60) becomes

$$E\left\{\frac{\partial}{\partial \hat{\underline{h}}_k^T[m]}\left[\frac{\partial J_{\mathrm{f}}[m+1]}{\partial \hat{\underline{h}}_k^*(m)}\right]\right\} \approx \frac{1}{2}\mathcal{W}_{M\times 2M}^{10}\mathcal{P}_k[m+1]\mathcal{W}_{2M\times M}^{10}, \tag{8.62}$$

where

$$\mathcal{P}_k[m+1] = \sum_{i=0,i\neq k}^{L} E\left\{\mathcal{D}_{x_i}^*[m+1]\mathcal{D}_{x_i}[m+1]\right\},\ \ k = 0, 1, \cdots, L.$$

Substituting (8.57) and (8.62) into (8.58) and multiplying by $\mathcal{W}_{2M\times M}^{10}$ produces the *constrained* NMCFLMS algorithm:

$$\begin{aligned}
&\underline{\hat{\boldsymbol{h}}}_k^{10}[m+1] \\
&\quad = \underline{\hat{\boldsymbol{h}}}_k^{10}[m] - 2\mu_{\mathrm{f}}\mathcal{W}_{2M\times M}^{10}\left\{\mathcal{W}_{M\times 2M}^{10}\mathcal{P}_k[m+1]\mathcal{W}_{2M\times M}^{10}\right\}^{-1}\cdot \\
&\qquad\qquad \mathcal{W}_{M\times 2M}^{10}\sum_{i=0}^{L}\mathcal{D}_{x_i}^{*}[m+1]\underline{\boldsymbol{e}}_{ik}^{01}[m+1] \\
&\quad = \underline{\hat{\boldsymbol{h}}}_k^{10}[m] - 2\mu_{\mathrm{f}}\mathcal{W}_{2M\times 2M}^{10}\mathcal{P}_k^{-1}[m+1]\cdot \\
&\qquad\qquad \sum_{i=0}^{L}\mathcal{D}_{x_i}^{*}[m+1]\underline{\boldsymbol{e}}_{ik}^{01}[m+1],
\end{aligned} \tag{8.63}$$

where

$$\begin{aligned}
\underline{\hat{\boldsymbol{h}}}_k^{10}[m] &= \mathcal{W}_{2M\times M}^{10}\underline{\hat{\boldsymbol{h}}}_k[m] = \mathbf{F}_{2M\times 2M}\begin{bmatrix}\hat{\mathbf{h}}_k^T[m] & \mathbf{0}\end{bmatrix}^T, \\
\underline{\boldsymbol{e}}_{ik}^{01}[m+1] &= \mathcal{W}_{2M\times M}^{01}\underline{\boldsymbol{e}}_{ik}[m+1] = \mathbf{F}_{2M\times 2M}\begin{bmatrix}\mathbf{0} & \mathbf{e}_{ik}^T[m+1]\end{bmatrix}^T, \\
\mathcal{W}_{2M\times 2M}^{10} &= \mathcal{W}_{2M\times M}^{10}\mathcal{W}_{M\times 2M}^{10} \\
&= \mathbf{F}_{2M\times 2M}\begin{bmatrix}\mathbf{I}_{M\times M} & \mathbf{0}_{M\times M} \\ \mathbf{0}_{M\times M} & \mathbf{0}_{M\times M}\end{bmatrix}\mathbf{F}_{2M\times 2M}^{-1},
\end{aligned}$$

and the relation

$$\begin{aligned}
\mathcal{W}_{2M\times M}^{10}\left\{\mathcal{W}_{M\times 2M}^{10}\mathcal{P}_k[m+1]\mathcal{W}_{2M\times M}^{10}\right\}^{-1}\mathcal{W}_{M\times 2M}^{10} &= \\
\mathcal{W}_{2M\times 2M}^{10}\mathcal{P}_k^{-1}&[m+1]
\end{aligned}$$

can be justified by post-multiplying both sides of the expression by $\mathcal{P}_k[m+1]\mathcal{W}_{2M\times M}^{10}$ and recognizing that $\mathcal{W}_{2M\times 2M}^{10}\mathcal{W}_{2M\times M}^{10} = \mathcal{W}_{2M\times M}^{10}$.

If the matrix $2\mathcal{W}_{2M\times 2M}^{10}$ is approximated by the identity matrix similar to (8.61) for $\mathcal{W}_{2M\times 2M}^{01}$, we finally deduce the *unconstrained* NMCFLMS algorithm:

$$\underline{\hat{\boldsymbol{h}}}_k^{10}[m+1] = \underline{\hat{\boldsymbol{h}}}_k^{10}[m] - \mu_{\mathrm{f}}\mathcal{P}_k^{-1}[m+1]\sum_{i=0}^{L}\mathcal{D}_{x_i}^{*}[m+1]\underline{\boldsymbol{e}}_{ik}^{01}[m+1], \tag{8.64}$$

where the normalization matrix $\mathcal{P}_k[m+1]$ is diagonal and it is easy to find its inverse. Again, the unit-norm constraint will be enforced on the modeling filter coefficients in the time domain.

Detailed implementation of the aforementioned adaptive multichannel (AMC) algorithms can be found in [32].

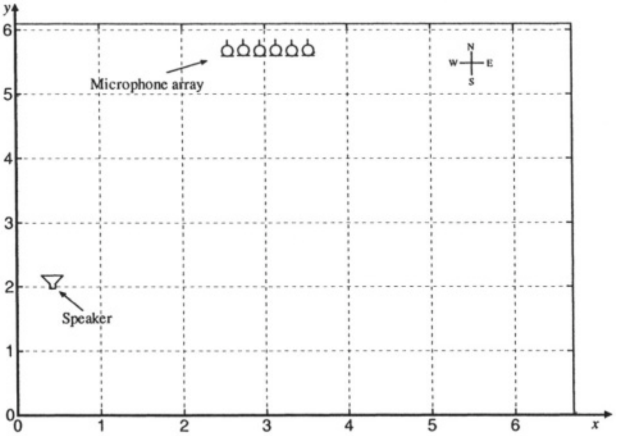


Figure 8.3 Varechoic chamber floor plan (coordinate values measured in meters); the loudspeaker source is located at (0.337, 2.162, 1.600); six microphones are placed at (2.437, 5.600, 1.400), (2.537, 5.600, 1.400), (2.637, 5.600, 1.400), (2.737, 5.600, 1.400), (2.837, 5.600, 1.400), (2.937, 5.600, 1.400), respectively.

7. EXPERIMENTS

7.1 EXPERIMENTAL SETUP

The measurements used in this chapter were made in the Varechoic chamber at Bell Labs [45]. A diagram of the floor plan layout is shown in Fig. 8.3. For convenience, positions in the floor plan will be designated by (x, y) coordinates with reference to the southwest corner and corresponding to meters along the (South, West) walls. The chamber is of size 6.7m × 6.1m × 2.9m$(x \times y \times z)$ with 368 electronically controlled panels that vary the acoustic absorption of the walls, floor, and ceiling [46]. Each panel consists of two perforated sheets whose holes, if aligned, expose sound absorbing material behind, but if shifted to misalign, form a highly reflective surface. The panels are individually controlled so that the holes on one particular panel are either fully open (absorbing state) or fully closed (reflective state). Therefore, by varying the binary state of each panel in any combination, 2^{238} different room characteristics can be simulated. A linear microphone array with six omni-directional microphones was employed in the measurement and the spacing between adjacent microphones is 10 cm. The array was mounted 1.4 m above the floor and parallel to the North wall at a distance of 50 cm. The six microphone positions are denoted as M1 (2.437, 5.600, 1.400), M2 (2.537, 5.600, 1.400), M3 (2.637, 5.600, 1.400), M4 (2.737, 5.600, 1.400), M5 (2.837, 5.600, 1.400), and M6 (2.937, 5.600, 1.400), respectively. The source was simulated by placing a loudspeaker at (0.337, 2.162, 1.600). The transfer functions of the acoustic channels between the loudspeaker and six microphones were measured at a 48 kHz sampling

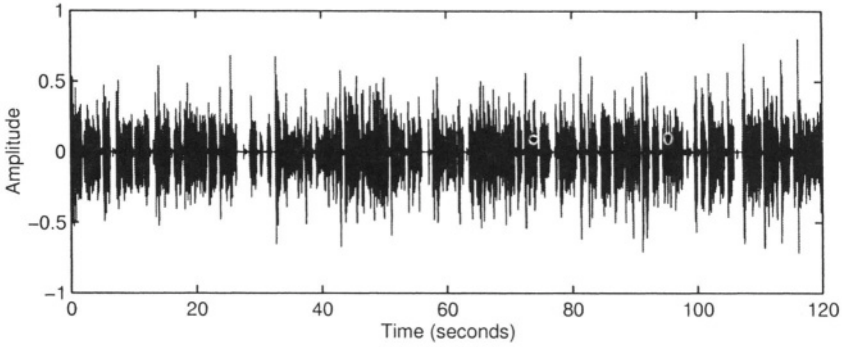


Figure 8.4 Speech signal used as the source, sampled at 16 kHz.

rate. Then the obtained channel impulse responses were downsampled to a 16 kHz sampling rate and truncated to 4096 samples. These measured impulse responses will be treated as the actual impulse responses in the TDE experiments.

The source signal is a sequence of a clean speech (from a female speaker) sampled at 16 kHz and of duration 2 minutes. The signal waveform is shown in Fig. 8.4. The multichannel system output is computed by convolving the speech source with the corresponding measured channel impulse responses and adding zero-mean, white, Gaussian noise to each one of these outputs for a given signal-to-noise ratio (SNR).

7.2 PERFORMANCE MEASURE

To better evaluate the performance of a time delay estimator, it would be helpful to classify an estimate as either “success” or “failure” [18, 29]. An estimate $\hat{\tau}_i$ for which the absolute error $|\hat{\tau}_i - \tau_i|$ exceeds $\mathrm{T_c}/2$, where $\mathrm{T_c}$ is the signal correlation time, and τ_i the true delay, is identified as a failure (or anomaly), which follows the terminology used in [29]. Otherwise, an estimate would be deemed as a success (or nonanomaly). In this chapter, $\mathrm{T_c}$ is defined as the width of the main lobe of the source signal autocorrelation function (taken between the – 3 dB points). For the particular source signal used here, which is sampled at 16 kHz, $\mathrm{T_c}$ is equal to 4.7 samples.

After time delay estimates are classified into the two classes, the TDE performance is evaluated in terms of the percentage of anomalies over the total estimates, and the MSE of the nonanomalous estimates.

The GCC (Section 3) and AED (Section 5) algorithms, using signals received by two sensors, estimate one delay, which is the TDOA between the two sensors (we assume they are Sensor 0 and 1). The multichannel cross-correlation (MCC) algorithm (Section 4.2 and 4.3), although exploring multiple sensors, also generates one time delay estimate (without loss of generality, we assume that is the

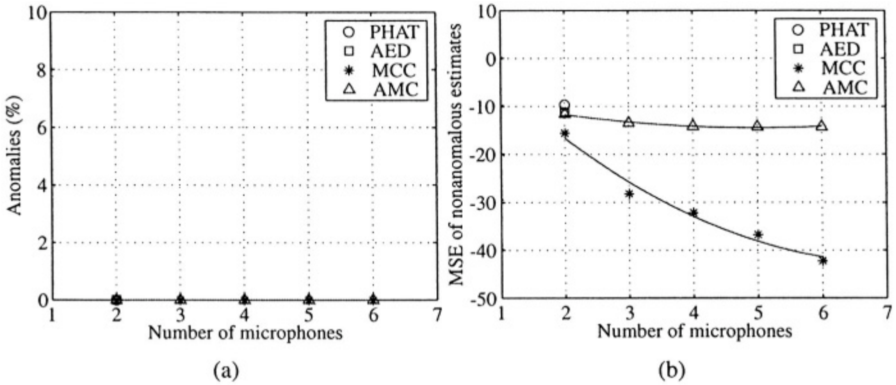


Figure 8.5 The TDE performances of the PHAT, AED, MCC, and AMC algorithms; $T_{60} = 240$ ms, SNR = 10 dB.

TDOA between Sensor 0 and 1). For the adaptive multichannel (AMC) TDE algorithm [the *unconstrained* NMCFLMS algorithm (Section 6.3)] where more than two channels are available, a time delay estimate for each sensor pair will be achieved. For the purpose of a fair comparison among algorithms, we only evaluate the time delay between Sensor 0 and 1, and delay estimates between other sensor pairs will be neglected.

7.3 EXPERIMENTAL RESULTS

For brevity, we cite four sets of experimental results. The first one involves a set of data obtained in a light reverberant and less noisy environment where the reverberation time, T_{60} (defined as the time for the sound to die away to a level 60 decibels below its original level and measured by Schroeder's method [47]), is approximately 240 ms, and SNR = 10 dB. The TDE results are presented in Fig. 8.5. As seen, all the four algorithms can accurately determine the relative TDOA with no anomalies being observed in this situation. For the two-microphone case, the four algorithms yield similar MSE. When more than two microphones are employed, the MSE of both the MCC and AMC algorithms reduces, showing the advantage of using multiple microphones. (In this and following figures, the fitting curve is a second order polynomial.)

The second experiment pertains to a set of data acquired in a condition where the reverberation is the same as in the previous setup, but this time the noise is much stronger with an SNR = –5 dB. The result is graphically portrayed in Fig. 8.6. It is noticed that all four algorithms deteriorate in their performance. Since the SNR is very low in this case, noise is the dominant distortion source that causes the performance degradation. From Fig. 8.6 (a), one can see that in the two-sensor case, the cross correlation based algorithms perform slightly better than the blind channel identification based approaches, suggesting that

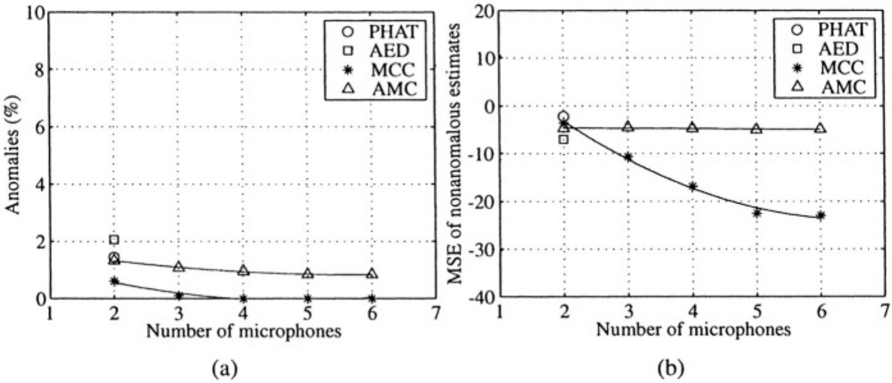
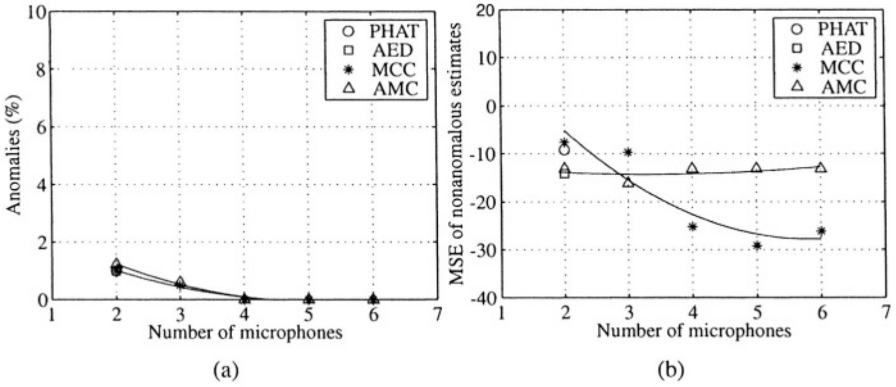


Figure 8.6 The TDE performance of the PHAT, AED, MCC, and AMC algorithms; $T_{60} = 240$ ms, SNR $= -5$ dB.

the cross correlation based methods are more tolerant to strong noise. Again, the TDE performance of both the MCC and AMC algorithms improves with the number of sensors.

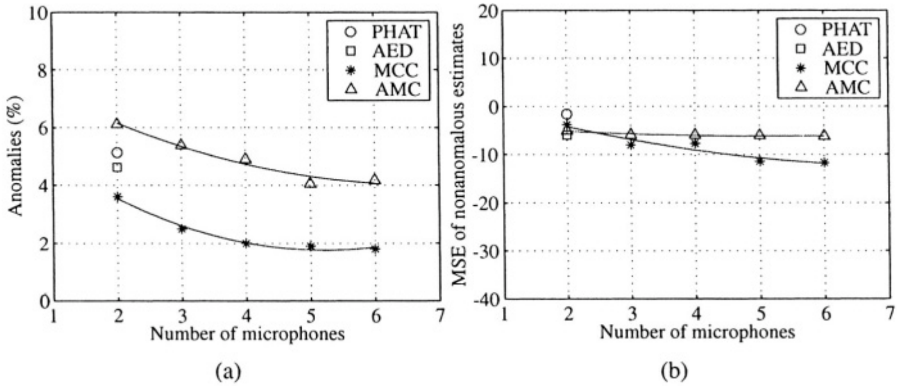
When reverberation becomes heavier, each microphone sensor receives a greater number of delayed and attenuated replicas of the source signal due to reflection of room boundaries, which makes the TDE problem harder. The third experiment considers a stronger reverberant but less noisy environment where $T_{60} = 580$ ms and SNR = 10 dB. The TDE result is shown in Fig. 8.7. Comparing Fig. 8.7 with Fig. 8.5, one can see that all the studied algorithms suffer performance degradation when reverberation increases. It is noticed from Fig. 8.7 (b) that in the two-microphone case, the blind channel identification based techniques have lower MSE compared to the cross correlation based algorithms, indicating that the former techniques are more robust with respect to strong reverberation.

The final experiment shows the TDE performance behavior in a heavily reverberant and strongly noisy environment. As seen from Fig. 8.8, all four algorithms suffer dramatic degradation in their performance. However, as the number of sensors increases, the performance of both the MCC and AMC algorithms improves. It is noted that both the MCC and AMC algorithms can greatly benefit from the use of multiple sensors. Among these two, the MCC algorithm improves faster in its performance with the number of microphones than does the AMC method. This is because the former exploits multiple microphones to estimate only one delay, which fully utilizes the redundancy provided by the array, while the latter technique estimates the TDOAs between each microphone pair. Since the MCC algorithm takes advantage of the array geometry to improve its performance, the array has to be well designed and calibrated. For the AMC algorithm, however, the TDOA between each sensor



(a) (b)

Figure 8.7 The TDE performances of the PHAT, AED, MCC, and AMC algorithms; $T_{60} = 580$ ms, SNR = 10 dB.



(a) (b)

Figure 8.8 The TDE performance of the PHAT, AED, MCC, and AMC algorithms; $T_{60} = 580$ ms, SNR = −5 dB.

pair is estimated based on identifying their impulse responses, so the array geometry is not of great concern.

8. CONCLUSIONS

Time delay estimation (TDE) in a reverberant acoustic environment still remains a very challenging and difficult problem. There are mainly two approaches to deal more efficiently with reverberation. The first is to use more than two sensors and take advantage of the redundancy. The second is the inclusion of the acoustic channel impulse responses into the TDE algorithms. This chapter reviewed some recent efforts in developing TDE techniques that are robust against reverberation. Addressed were the generalized cross-correlation (GCC) method, the multichannel cross-correlation (MCC) technique, the adaptive eigenvalue decomposition (AED) algorithm, and the adaptive multichannel (AMC) algorithms.

The GCC method is a two-sensor technique based upon the ideal single-path acoustic propagation model. It performs fairly well in moderate noise and reverberation conditions when the two prefilters are properly selected. However, it suffers severe performance degradation in the presence of strong noise and heavy reverberation. The MCC algorithm is a natural generalization of the classical cross-correlation method to the multichannel case. It takes advantage of the redundancy provided by multiple sensors to estimate one time delay. The TDE performance of the MCC algorithm in the presence of noise and reverberation improves with the number of microphone sensors. The AED algorithm is also a two-sensor technique. Different from the GCC method, it is based on the reverberant propagation model and obtains the delay estimates based on blindly identifying the two impulse responses. The multichannel adaptive algorithm is an extension of the AED approach, which improves time delay estimation by exploiting the diversity among multiple channels.

Experiments were performed using data measured in the Varechoic chamber at Bell Labs. It was shown that in the two-microphone case, the correlation based techniques are more robust with respect to noise, while more sensitive to reverberation than the blind channel identification based algorithms. When multiple microphone sensors are available, both the MCC and AMC algorithms improve the TDE performances with the number of microphones, with the former coping better than the latter against reverberation and noise. However, to make the MCC algorithm efficient, the microphone array has to be well designed and calibrated, which is not of big concern for the AMC method since it estimates TDOA between each microphone pair by identifying the channel impulse responses.

References

[1] J. E. Ehrenberg, T. E. Ewatt, and R. D. Morris, "Signal processing techniques for resolving individual pulses in multipath signal," *J. Acoust. Soc. Am.,* vol. 63, pp. 1861–1865, June 1978.

[2] N. L. Owsley and G. R. Swope, "Time delay estimation in a sensor array," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-27, pp. 519–523, June 1981.

[3] R. J. Tremblay, G. C. Carter, and D. W. Lytle, "A practical approach to the estimation of amplitude and time-delay parameters of a composite signal," *IEEE J. Oceanic Eng.,* vol. OE-12, pp. 273–278, Jan. 1987.

[4] R. Wu, J. Li, and Z. S. Liu, "Super resolution time delay estimation via MODE-WRELAX," *IEEE Trans. Aerosp. Electron. Syst.,* vol. 35, pp. 294–307, Jan. 1999.

[5] C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-24, pp. 320–327, Aug. 1976.

[6] G. C. Carter, 'Time delay estimation for passive sonar signal processing," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-29, pp. 463–470, June 1981.

[7] G. C. Carter, "Coherence and time delay estimation," in *Signal Processing Handbook,* C. H. Chen, Ed. Marcel Dekker, Inc., New York, 1988.

[8] A. H. Quazi, "An overview on the time delay estimation in active and passive systems for target localization," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-29, pp. 527–533, June 1981.

[9] G. C. Carter, Ed., *Coherence and time delay estimation: an applied tutorial for research, development, test and evaluation engineers,* IEEE Press, 1993.

[10] M. Feder and E. Weinstein, "Parameter estimation of superimposed signals using the EM algorithm," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-36, pp. 477–489, Apr. 1988.

[11] G. Su and M. Morf, "The signal subspace approach for multiple wide-band emitter location," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-31, pp. 1502–1522, Dec. 1983.

[12] S. S. Reddy, "Multiple source location – a digital approach," *IEEE Trans. Aerosp. Electron. Syst.,* vol. AES-15, pp. 95–105, Jan. 1979.

[13] T. G. Manickam, R. J. Vaccaro, and D. W. Tufts, "A least-squares algorithm for multipath time-dealy estimation," *IEEE Trans. Signal Processing,* vol. 42, pp. 3229–3233, Nov. 1994.

[14] J.-J. Fuchs, "Multipath time-delay detection and estimation," *IEEE Trans. Signal Processing,* vol. 47, pp. 237–243, Jan. 1999.

[15] J. Benesty, "Adaptive eigenvalue decomposition algorithm for passive acoustic source localization," *J. Acoust. Soc. Am.,* vol. 107, pp. 384–391, Jan. 2000.

[16] J. C. Hassab and R. E. Boucher, "Performance of the generalized cross correlator in the presence of a strong spectral peak in the signal," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-29, pp. 549–555, June 1981.

[17] L. E. Miller and J. S. Lee, "Error analysis of time delay estimation using a finite integration time correlator," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-29, pp. 490–496, June 1981.

[18] J. P. Ianniello, "Time delay estimation via cross-correlation in the presence of large estimation errors," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-30, pp. 998–1003, Dec. 1982.

[19] M. Azaria and D. Hertz, "Time delay estimation by generalized cross correlation methods," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-32, pp. 280–285, Apr. 1984.

[20] Y. Bar-Shalom, F. Palmieri, A. Kumar, and H. M. Shertukde, "Analysis of wide-band cross correlation for time-delay estimation," *IEEE Trans. Signal Processing,* vol. 41, pp. 385–387, Jan. 1993.

[21] J. K. Tugnait, "Time delay estimation with unknown spatially correlated Gaussian noise," *IEEE Trans. Signal Processing,* vol. 41, pp. 549–558, Feb. 1993.

[22] Y. Wu, "Time delay estimation of non-Gaussian signal in unknown Gaussian noises using third-order cumulants," *Electron. Lett.,* vol. 38, pp. 930–931, Aug. 2002.

[23] Y. Huang and J. Benesty, "A class of frequency-domain adaptive approaches to blind multichannel identification," *IEEE Trans. Signal Processing,* vol. 51, pp. 11–24, Jan. 2003.

[24] F. A. Reed, P. L. Feintuch, and N. J. Bershad, "Time delay estimation using the LMS adaptive filter–static behavior," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-29, pp. 561–571, June 1981.

[25] D. M. Etter and S. D. Stearns, "Adaptive estimation of time delays in sampled data systems," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-29, pp. 582–587, June 1981.

[26] D. H. Youn, N. Ahmed, and G. C. Carter, "On using the LMS algoritm for time delay estimation," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-30, pp. 798–801, Oct. 1982.

[27] P. C. Ching and Y. T. Chan, "Adaptive time delay estimation with constraints," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-36, pp. 599–602, Apr. 1988.

[28] H. C. So, P. C. Ching, and Y. T. Chan, "A new algorithm for explicit adaptation of time delay," *IEEE Trans. Signal Processing,* vol. 42, pp. 1816–1820, July 1994.

[29] B. Champagne, S. Bédard, and A. Stéphenne, "Performance of time-delay estimation in the presence of room reverberation," *IEEE Trans. Speech Audio Processing,* vol. 4, pp. 148–152, Mar. 1996.

[30] B. G. Ferguson, "Improved time-delay estimates of underwater acoustic signals using beamforming and prefiltering techniques," *IEEE J. Oceanic Eng.,* vol. OE-14, pp. 238–244, July 1989.

[31] S. M. Griebel and M.S. Brandstein, "Microphone array source localization using realizable delay vectors," in *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics,* 2001.

[32] Y. Huang and J. Benesty, "Adaptive multichannel time delay estimation based on blind system identification for acoustic source localization," in *Adaptive Signal Processing–Applications to Real-World Problems,* J. Benesty and Y. Huang, Eds., Springer, New York, 2003.

[33] P. P. Moghaddam, H. Amindavar, and R. L. Kirlin, "A new time-delay estimation in multipath," *IEEE Trans. Signal Processing,* vol. 51, pp. 1129–1142, May 2003.

[34] J. P. Ianniello, "Large and small error performance limits for multipath time delay estimation," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-34, pp. 245–251, Apr. 1986.

[35] J. C. Hassab, "Contact localization and motion analysis in the ocean environment: a perspective," *IEEE J. Oceanic Eng.,* vol. OE-8, pp. 136–147, July 1983.

[36] F. El-Hawary, F. Aminzadeh, and G. A. N. Mbamalu, "The generalized Kalman filter approach to adpative underwater target tracking," *IEEE J. Oceanic Eng.,* vol. OE-17, pp. 129–137, Jan. 1992.

[37] C. S. Clay and H. Medwin, *Acoustical Oceanography,* Wiley, New York, NY, 1977.

[38] G. C. Carter, A. H. Nuttall, and P. G. Cable, "The smoothed coherence transform," *Proc. IEEE,* vol. 61, pp. 1497–1498, Oct. 1973.

[39] P. R. Roth, "Effective measurements using digital signal analysis," *IEEE Spectrum,* vol. 8, pp. 62–70, Apr. 1971.

[40] S. Haykin, "Radar array processing for angle of arrival estimation," in *Array Signal Processing,* S. Haykin, Ed., pp. 194–292. Prentice-Hall, 1985.

[41] G. H. Golub and C. F. Van Loan, *Matrix Computations,* John Hopkins University Press, Baltimore, MD, 1996.

[42] Y. Huang, J. Benesty, and G. W. Elko, "Adaptive eigenvalue decomposition algorithm for realime acoustic source localization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing,* 1999, pp. 937–940.

[43] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel indentification," *IEEE Trans. Signal Processing,* vol. 43, pp. 2983–2993, Dec. 1995.

[44] Y. Huang and J. Benesty, "Adaptive multi-channel least mean square and Newton algorithms for blind channel identification," *Signal Processing,* vol. 82, pp. 1127–1138, Aug. 2002.

[45] A. Härmä, "Acoustic measurement data from the varechoic chamber," *Technical Memorandum, Agere Systems,* Nov. 2001.

[46] W. C. Ward, G. W. Elko, R. A. Kubli, and W. C. McDougald, "The new Varechoic chamber at AT&T Bell Labs," in *Proc. Wallace Clement Sabine Centennial Symposium,* 1994, pp. 343–346.

[47] M. R. Schroeder, "New method for measuring reverberation," *J. Acoust. Soc. Am.,* vol. 37, 1965.

This page intentionally left blank

Chapter 9

SOURCE LOCALIZATION

Yiteng (Arden) Huang
Bell Laboratories, Lucent Technologies
arden@research.bell-labs.com

Jacob Benesty
Université du Québec, INRS-EMT
benesty@inrs-emt.uquebec.ca

Gary W. Elko
Avaya Labs, Avaya
gwe@avaya.com

Abstract Time delays of arrival are important parametric representations of acoustic signals captured by a passive microphone array. But they are rarely directly used in an array signal processing system that usually assumes the explicit knowledge of a sound source location. In this chapter we turn our attention to passive source localization techniques that extract the spatial information of a sound source from time delays of arrival estimated using approaches investigated in the previous chapter. Parametric estimation is in general difficult when the signal model is nonlinear. Such is the case in the problem of source localization. We will review a number of advanced approaches with some examples and comment on their technical merits and shortcomings for practical implementations. A successful real-time acoustic source localization system for video camera steering in teleconferencing is presented at the end of this chapter.

Keywords: Source Localization, Estimation Theory, Least Squares, Lagrange Multiplier, Measurement Error, Real-Time Implementation

1. INTRODUCTION

As precise context understanding helps a reader to comprehend the correct connotation that a message carries, meticulous spatial perception makes it a lot easier for a listener to grasp the gist and even the implication of a conversation that he or she is involved, particularly when there are multiple participants. While the former has been well documented and accepted, the latter is not given adequate attention mainly because we make no conscious effort to localize a sound source. However, as a matter of fact, the knowledge of environmental sound sources and the capability of tracking them are essential to natural conversations and collaborations that are pursued in the next-generation multimedia telecommunication systems.

Locating radiative point sources using passive, stationary sensor arrays is of considerable interest and has been a repeated theme of research in radar [1], [2], underwater sonar [3], and seismology [4]. A common method is to have the estimate of source location based on time delay of arrival (TDOA) measurements between distinct sensor pairs. Nowadays, the same kind of techniques is used to localize and track acoustic sources for emerging applications such as automatic camera tracking for video-conferencing [5], [6], [7], [8] and beamformer steering for suppressing noise and reverberation [9], [10], [11], [12] in all types of communication and voice processing systems. We believe that such a time delay estimation (TDE) based method will continue playing an important role in tomorrow's multimedia communication systems.

In order to estimate the location of a single sound source using estimated TDOAs, one needs to first choose a data model which describes how a source location is related to TDOA observations and how noise or measurement error is introduced. If errors (that are possibly mutually dependent) are supposed to be additive to and independent of the TDOA measurements, the source would be located at the intersection of a set of hyperboloids. Finding this intersection is a nonlinear problem. Although such an additive model does not easily lend itself to modification due to the nonlinearity, it describes the principal constraints imposed by the TDOA data in a simple way and thus is widely used in studying the source localization problem.

There is a rich literature of source localization techniques that use the additive measurement error model. Important distinctions between these methods include likelihood-based versus least-squares and linear approximation versus direct numerical optimization, as well as iterative versus closed-form algorithms.

In early research of source localization with passive sensor arrays, the maximum likelihood (ML) principle was widely utilized [13], [14], [15], [16] because of the proven asymptotic consistency and efficiency of an ML estimator (MLE). However, the number of microphones in an array for camera pointing or beamformer steering in multimedia communication systems is always limited, which makes acoustic source localization a finite-sample rather than a large-sample problem. Moreover, ML estimators require additional assumptions about the distributions of the measurement errors. One approach is to invoke the central limit theorem and assumes a Gaussian approximation, which makes the likelihood function easy to formulate. Although a Gaussian error was justified by Hahn and Tretter [13] for continuous-time processing, it can be difficult to verify and the MLE is no longer optimal when sampling introduces additional errors in discrete-time processing. To compute the solution to the MLE, a linear approximation and iterative numerical techniques have to be used because of the nonlinearity of the hyperbolic equations. The Newton-Raphson iterative method [17], the Gauss-Newton method [18], and the least-mean-square (LMS) algorithm are among possible choices. But for these iterative approaches, selecting a good initial guess to avoid a local minimum is difficult and convergence to the optimal solution cannot be guaranteed. Therefore, it is our opinion that an ML-based estimator is not suitable for the real-time implementation of a source localization system.

For real-time applications, closed-form estimators are desired and appropriately, have also gained wider attention. Of the closed-form estimators, triangulation is the most straightforward [6]. However, with triangulation it is difficult to take advantage of extra sensors and the TDOA redundancy. Nowadays most closed-form algorithms exploit a least-squares principle, which makes no additional assumption about the distribution of measurement errors. To construct a least-squares estimator, one needs to define an error function based on the measured TDOAs. Different error functions will result in different estimators with different complexity and performance. Schmidt [19] showed that the TDOAs to three sensors whose positions are known provide a straight line of possible source locations in two dimensions and a plane in three dimensions. By intersecting the lines/planes specified by different sensor triplets, he obtained an estimator called plane intersection. Another closed-form estimator, termed spherical intersection (SX), employed a spherical LS criterion [20]. The SX algorithm is mathematically simple, but requires an *a priori* solution for the source range, which may not exist or may not be unique in the presence of measurement errors. Based on the same criterion, Smith and Abel [21] proposed the spherical interpolation (SI) method, which also solved for the source range, again in the LS sense. Although the SI method has less bias, it is not efficient and it has a large standard deviation relative to the Cramèr-Rao lower bound (CRLB). With the SI estimator, the source range is a byproduct that is assumed to be independent of the location coordinates. Chan and Ho [22] improved the SI estimation with a second LS estimator that accommodates the information redundancy from the SI estimates and updates the squares of the coordinates, We shall refer to this method as the quadratic-correction least-squares (QCLS) approach. In the QCLS estimator, the covariance matrix of measurement errors is used. But this information can be difficult to properly assume or accurately estimate, which results in a performance degradation in practice. When the SI estimate is analyzed and the quadratic correction is derived in the QCLS estimation procedure, perturbation approaches are employed and, presumptively, the magnitude of measurement errors has to be small. It has been indicated in [23] that the QCLS estimator yields an unbiased solution with a small standard deviation that is close to the CRLB at a moderate noise level. But when noise is practically strong, its bias is considerable and its variance could no longer approach the CRLB according to our Monte-Carlo simulations. Recently a linear-correction least-squares (LCLS) algorithm has been proposed by the authors in [24]. This method applies the additive measurement error model and employs the technique of Lagrange multipliers. It makes no assumption on the covariance matrix of measurement errors and utilizes no linear approximation that holds only in the case of small perturbation.

In the following, we will develop a comprehensive framework for investigating the problem of source localization and will comparatively study a number of approaches. We will evaluate all these algorithms with respect to estimation accuracy and efficiency, computational complexity, implementation flexibility, and adaptation capabilities to different and varying environments.

2. SOURCE LOCALIZATION PROBLEM

The problem addressed here is the determination of the location of an acoustic source given the array geometry and the relative TDOA measurements among different microphone pairs. The problem can be stated mathematically as follows.

The array consists of $N + 1$ microphones located at positions

$$\mathbf{r}_i \triangleq \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^T, \ i = 0, ..., N \tag{9.1}$$

in Cartesian coordinates (see Fig. 9.1), where $(\cdot)^T$ denotes transpose of a vector or a matrix. The first microphone ($i = 0$) is regarded as the reference and is placed at the origin of the coordinate system, i.e. $\mathbf{r}_0 = [0, 0, 0]^T$. The acoustic source is located at $\mathbf{r}_\mathrm{s} \triangleq [x_\mathrm{s}, y_\mathrm{s}, z_\mathrm{s}]^T$. The distances from the origin to the i-th microphone and the source are denoted by R_i and R_s, respectively, where

$$R_i \triangleq \|\mathbf{r}_i\| = \sqrt{x_i^2 + y_i^2 + z_i^2}, \ \ i = 1, ..., N \tag{9.2}$$

$$R_\mathrm{s} \triangleq \|\mathbf{r}_\mathrm{s}\| = \sqrt{x_\mathrm{s}^2 + y_\mathrm{s}^2 + z_\mathrm{s}^2}. \tag{9.3}$$

The distance between the source and the i-th microphone is denoted by

$$D_i \triangleq \|\mathbf{r}_i - \mathbf{r}_\mathrm{s}\| = \sqrt{(x_i - x_\mathrm{s})^2 + (y_i - y_\mathrm{s})^2 + (z_i - z_\mathrm{s})^2}\,. \tag{9.4}$$

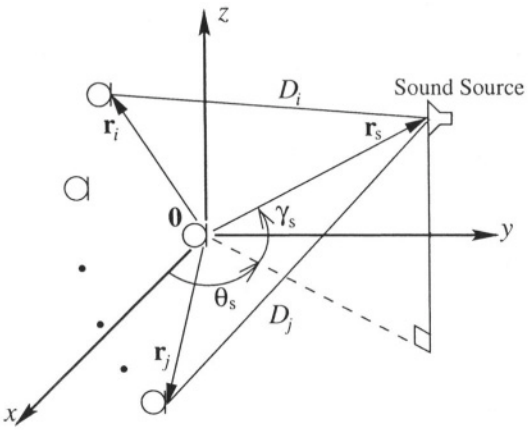


Figure 9.1 Spatial diagram illustrating variables defined in the source localization problem.

The difference in the distances of microphones i and j from the source is given by

$$d_{ij} \triangleq D_i - D_j, \quad i, j = 0, ..., N. \tag{9.5}$$

This difference is usually termed the *range difference.* It is proportional to the time delay of arrival τ_{ij}. If the speed of sound is c, then

$$d_{ij} = c \cdot \tau_{ij}. \tag{9.6}$$

The speed of sound (in m/s) can be estimated from the air temperature t_{air} (in degrees Celsius) according to the following approximate (first-order) formula,

$$c \approx 331 + 0.610 \times t_{\text{air}} . \tag{9.7}$$

The localization problem is then to estimate $\mathbf{r}_s$ given the set of $\mathbf{r}_i$ and τ_{ij}. Note that there are $(N + 1)N/2$ distinct TDOA estimates τ_{ij}, which exclude the case $i = j$ and count the $\tau_{ij} = -\tau_{ji}$ pair only once. However, in the absence of noise, the space spanned by these TDOA estimates is N-dimensional. Any N linearly independent TDOAs determine all of the others. In a noisy environment, the TDOA redundancy can be used to improve the accuracy of the source localization algorithms, but this would increase their computational complexity. For simplicity and also without loss of generality, we choose $\tau_{i0}, i = 1, ..., N$ as the basis for this $\mathbb{R}^N$ space in this chapter.

3. MEASUREMENT MODEL AND CRAMÈR-RAO LOWER BOUND FOR SOURCE LOCALIZATION

When the source localization problem is examined using estimation theory, the measurements of the range differences are modeled by:

$$d_{i0} = g_i(\mathbf{r}_s) + \epsilon_i,\ i = 1, ..., N \tag{9.8}$$

where,

$$g_i(\mathbf{r}_s) = \|\mathbf{r}_i - \mathbf{r}_s\| - \|\mathbf{r}_s\|,$$

and the ϵ_i's are measurement errors. In a vector form, such an additive measurement error model becomes,

$$\mathbf{d} = \mathbf{g}(\mathbf{r}_s) + \boldsymbol{\epsilon}, \tag{9.9}$$

where

$$\begin{aligned} \mathbf{d} &= \begin{bmatrix} d_{10} & d_{20} & \cdots & d_{N0} \end{bmatrix}^T, \\ \mathbf{g}(\mathbf{r}_s) &= \begin{bmatrix} g_1(\mathbf{r}_s) & g_2(\mathbf{r}_s) & \cdots & g_N(\mathbf{r}_s) \end{bmatrix}^T, \\ \boldsymbol{\epsilon} &= \begin{bmatrix} \epsilon_1 & \epsilon_2 & \cdots & \epsilon_N \end{bmatrix}^T. \end{aligned}$$

Further, we postulate that the additive measurement errors have mean zero and are independent of the range difference observation, as well as the source location $\mathbf{r}_s$. For a continuous-time estimator, the corrupting noise, as indicated in [13], is jointly Gaussian distributed. The probability density function (PDF) of $\mathbf{d}$ conditioned on $\mathbf{r}_s$ is subsequently given by,

$$p(\mathbf{d}|\mathbf{r}_s) = \frac{\exp\left\{-\frac{1}{2}\left[\mathbf{d} - \mathbf{g}(\mathbf{r}_s)\right]^T \mathbf{C}_{\boldsymbol{\epsilon}}^{-1}\left[\mathbf{d} - \mathbf{g}(\mathbf{r}_s)\right]\right\}}{\sqrt{(2\pi)^N \det(\mathbf{C}_{\boldsymbol{\epsilon}})}}, \tag{9.10}$$

where $\mathbf{C}_{\boldsymbol{\epsilon}}$ is the covariance matrix of $\boldsymbol{\epsilon}$ and "det" denotes the determinant. Note that $\mathbf{C}_{\boldsymbol{\epsilon}}$ is independent of $\mathbf{r}_s$ by assumption. Since digital equipment is used to sample the microphone waveforms and estimate the TDOAs, the error introduced by discrete-time processing also has to be taken into account. When this is done, the measurement error is no longer Gaussian and is more properly modeled as a mixture of a Gaussian noise and a noise that is uniformly distributed over $[-T_s c/2, T_s c/2]$, where T_s is the sampling period. As an example, for a digital source location estimator with an 8 KHz sampling rate operating at room temperature (25 degrees Celsius, i.e. $c \approx 346.25$ meters per second), the maximum error in range difference estimates due to sampling is about ±2.164 cm, which leads to considerable errors in the location estimate, especially when the source is far from the microphone array.

Under the measurement model (9.9), we are now faced with the parameter estimation problem of extracting the source location information from the mismeasured range differences or the equivalent TDOAs. For an *unbiased* estimator, a Cramèr-Rao lower bound (CRLB) can be placed on the variance of each estimated coordinate of the source location. However, since the range difference function $\mathbf{g}(\mathbf{r}_\mathrm{s})$ in the measurement model is nonlinear in the parameters under estimation, it is very difficult (or even impossible) to find an unbiased estimator that is mathematically simple and attains the CRLB. The CRLB is usually used as a benchmark against which the statistical efficiency of any unbiased estimators can be compared.

In general, without any assumptions made about the PDF of the measurement error $\boldsymbol{\epsilon}$, the CRLB of the i-th $(i = 1, 2, 3)$ parameter variance is found as the $[i, i]$ element of the inverse of the Fisher information matrix defined by [25]:

$$[\mathbf{I}(\mathbf{r}_\mathrm{s})]_{ij} \triangleq -E\left[\frac{\partial^2 \ln p(\mathbf{d}|\mathbf{r}_\mathrm{s})}{\partial r_{\mathrm{s},i}\partial r_{\mathrm{s},j}}\right], \tag{9.11}$$

where the three parameters of $\mathbf{r}_\mathrm{s}$, i.e., $r_{\mathrm{s},1}$, $r_{\mathrm{s},2}$, and $r_{\mathrm{s},y}$, are respectively x, y, and z coordinates of the source location.

In the case of a Gaussian measurement error, the Fisher information matrix turns into [22]

$$\mathbf{I}(\mathbf{r}_\mathrm{s}) = \left[\frac{\partial\mathbf{g}(\mathbf{r}_\mathrm{s})}{\partial\mathbf{r}_\mathrm{s}}\right]^T \mathbf{C}_{\boldsymbol{\epsilon}}^{-1} \left[\frac{\partial\mathbf{g}(\mathbf{r}_\mathrm{s})}{\partial\mathbf{r}_\mathrm{s}}\right], \tag{9.12}$$

where $\partial\mathbf{g}(\mathbf{r}_\mathrm{s})/\partial\mathbf{r}_\mathrm{s}$ is an $N \times 3$ Jacobian matrix defined as,

$$\frac{\partial\mathbf{g}(\mathbf{r}_\mathrm{s})}{\partial\mathbf{r}_\mathrm{s}} = \begin{bmatrix} \frac{\partial g_1(\mathbf{r}_\mathrm{s})}{\partial x_\mathrm{s}} & \frac{\partial g_1(\mathbf{r}_\mathrm{s})}{\partial y_\mathrm{s}} & \frac{\partial g_1(\mathbf{r}_\mathrm{s})}{\partial z_\mathrm{s}} \\ \frac{\partial g_2(\mathbf{r}_\mathrm{s})}{\partial x_\mathrm{s}} & \frac{\partial g_2(\mathbf{r}_\mathrm{s})}{\partial y_\mathrm{s}} & \frac{\partial g_2(\mathbf{r}_\mathrm{s})}{\partial z_\mathrm{s}} \\ \vdots & \vdots & \vdots \\ \frac{\partial g_N(\mathbf{r}_\mathrm{s})}{\partial x_\mathrm{s}} & \frac{\partial g_N(\mathbf{r}_\mathrm{s})}{\partial y_\mathrm{s}} & \frac{\partial g_N(\mathbf{r}_\mathrm{s})}{\partial z_\mathrm{s}} \end{bmatrix} = \begin{bmatrix} (\mathbf{u}_1 - \mathbf{u}_0)^T \\ (\mathbf{u}_2 - \mathbf{u}_0)^T \\ \vdots \\ (\mathbf{u}_N - \mathbf{u}_0)^T \end{bmatrix}, \tag{9.13}$$

and,

$$\mathbf{u}_i = \frac{\mathbf{r}_\mathrm{s} - \mathbf{r}_i}{\|\mathbf{r}_\mathrm{s} - \mathbf{r}_i\|} = \frac{\mathbf{r}_\mathrm{s} - \mathbf{r}_i}{D_i},\ i = 0, 1, ..., N \tag{9.14}$$

is the normalized vector of unit length pointing from the i-th microphone to the sound source.

4. MAXIMUM LIKELIHOOD ESTIMATOR

In the previous section, the measurement model for the source localization problem was investigated and the CRLB for any unbiased estimator was determined. Since the measurement model is highly nonlinear, an efficient estimator that attains the CRLB may not exist or might be impossible to find even if it does exist. In practice, the maximum likelihood estimator is the most popular approach. It has the well-proven advantage of asymptotic efficiency for a large sample space.

To apply the maximum likelihood principle, the statistical characteristics of the measurements need to be known or properly assumed prior to any processing. From the central limit theorem and also for mathematical simplicity, the measurement error is usually modeled as Gaussian and the likelihood function is given by (9.10), which is considered as a function of the source position $\mathbf{r}_\mathrm{s}$ under estimation.

Since the exponential function is monotonically increasing, the MLE is equivalent to minimizing a (log-likelihood) cost function defined as,

$$\mathcal{E}_{\mathrm{MLE}}(\mathbf{r}_\mathrm{s}) \triangleq [\mathbf{d} - \mathbf{g}(\mathbf{r}_\mathrm{s})]^T \, \mathbf{C}_\epsilon^{-1} \, [\mathbf{d} - \mathbf{g}(\mathbf{r}_\mathrm{s})] \,. \tag{9.15}$$

Direct estimation of the minimizer is generally not practical. If the noise at different microphones is assumed to be uncorrelated, the covariance matrix is diagonal:

$$\mathbf{C}_\epsilon = \mathrm{diag}(\sigma_1^2, \sigma_2^2, ..., \sigma_N^2), \tag{9.16}$$

where σ_i^2 $(i = 1, 2, ..., N)$ is the variance of ϵ_i, and the cost function (9.15) becomes,

$$\mathcal{E}_{\mathrm{MLE}}(\mathbf{r}_\mathrm{s}) = \sum_{i=1}^{N} \frac{[d_{i0} - g_i(\mathbf{r}_\mathrm{s})]^2}{\sigma_i^2}. \tag{9.17}$$

Among other approaches, the steepest descent algorithm can be used to find $\hat{\mathbf{r}}_{\mathrm{s,MLE}}$ iteratively with

$$\hat{\mathbf{r}}_\mathrm{s}(k+1) = \hat{\mathbf{r}}_\mathrm{s}(k) - \frac{1}{2}\mu \, \nabla \, \mathcal{E}_{\mathrm{MLE}}(\hat{\mathbf{r}}_\mathrm{s}(k)) \,, \tag{9.18}$$

where μ is the step size.

The foregoing MLE can be determined and is asymptotically optimal for this problem only if its two assumptions (Gaussian and uncorrelated measurement noise) hold. However, this is not the case in practice as discussed in Section 3. Furthermore, the number of microphones in an array for camera pointing or beamformer steering is always limited, which makes the source localization a finite-sample rather than a large-sample problem. In addition, the cost function (9.17) is generally not strictly concave. In order to avoid a local minimum with the steepest descent algorithm, we need to select a good initial guess of the source location, which is difficult to do in practice, and convergence of the iterative algorithm to the desired solution cannot be guaranteed.

5. LEAST SQUARES ESTIMATORS

Two limitations of the MLE are that probabilistic assumptions have to be made about the measured range differences and that the iterative algorithm to find the solution is computationally intensive. An alternative method is the well-known least squares estimator (LSE). The LSE makes no probabilistic assumptions about the data and hence can be applied to the source localization problem in which a precise statistical characterization of the data is hard to determine. Furthermore, an LSE usually produces a closed-form estimate that is desirable in real-time applications. In this section, we begin by investigating the least squares (LS) error criteria and then develop different LS approaches to source localization.

5.1 THE LEAST SQUARES ERROR CRITERIA

In the LS approach, we attempt to minimize a squared error function that is zero in the absence of noise and model inaccuracies. Different error functions can be defined for closeness from the assumed (noiseless) signal based on hypothesized parameters to the observed data. When these are applied, different LSEs will be derived. For the source localization problem two LS error criteria can be constructed and will be presented here.

5.1.1 Hyperbolic LS Error Function. The first LS error function is defined as the difference between the observed range difference and that generated by a signal model depending upon the unknown parameters. Such an error function is routinely used in many LS estimators

$$\mathbf{e}_{\mathrm{h}}(\mathbf{r}_{\mathrm{s}}) \overset{\triangle}{=} \mathbf{d} - \mathbf{g}(\mathbf{r}_{\mathrm{s}}), \tag{9.19}$$

and the corresponding LS criterion is given by

$$J_{\mathrm{h}} = \mathbf{e}_{\mathrm{h}}^T \mathbf{e}_{\mathrm{h}} = [\mathbf{d} - \mathbf{g}(\mathbf{r}_{\mathrm{s}})]^T [\mathbf{d} - \mathbf{g}(\mathbf{r}_{\mathrm{s}})]. \tag{9.20}$$

In the source localization problem, an observed range difference d_{i0} defines a hyperboloid in 3D space. All points lying on such a hyperboloid are potential source locations and all have the same range difference d_{i0} to the two microphones i and 0. Therefore, a sound source that is located by minimizing the hyperbolic LS error criterion (9.20) has the shortest distance to all hyperboloids associated with different microphone pairs and specified by the estimated range differences.

In (9.19), the signal model $\mathbf{g}(\mathbf{r}_{\mathrm{s}})$ consists of a set of hyperbolic functions. Since they are nonlinear, minimizing (9.20) leads to a mathematically intractable solution as N gets large. Moreover, the hyperbolic function is very sensitive to noise, especially for far-field sources. As a result, it is rarely used in practice.

When the statistical characteristics of the corrupting noise are unknown, uncorrelated *white* Gaussian noise is one reasonable assumption. In this case, it is not surprising that the hyperbolic LSE and the MLE minimize (maximize) similar criteria.

5.1.2 Spherical LS Error Function. The second LS criterion is based on the errors found in the distances from a hypothesized source location to the microphones. In the absence of measurement errors, the correct source location is preferably at the intersection of a group of spheres centered at the microphones. When measurement errors are present, the best estimate of the source location would be the point that yields the shortest distance to those spheres defined by the range differences and the hypothesized source range.

Consider the distance D_i from the i-th microphone to the source. From the definition of the range difference (9.5) and the fact that $D_0 = R_s$, we have:

$$\hat{D}_i = R_s + d_{i0}, \tag{9.21}$$

where $\hat{D}_i$ denotes an observation based on the measured range difference. From the inner product, we can derive the true value for D_i^2, the square of the noise-free distance generated by a spherical signal model

$$D_i^2 = \|\mathbf{r}_i - \mathbf{r}_s\|^2 = R_i^2 - 2\mathbf{r}_i^T\mathbf{r}_s + R_s^2. \tag{9.22}$$

The spherical LS error function is then defined as the difference between the measured and hypothesized values

$$\begin{aligned} e_{sp,i}(\mathbf{r}_s) &\triangleq \frac{1}{2}\left(\hat{D}_i^2 - D_i^2\right) \\ &= \mathbf{r}_i^T\mathbf{r}_s + d_{i0}R_s - \frac{1}{2}(R_i^2 - d_{i0}^2), \quad i = 1, ..., N. \end{aligned} \tag{9.23}$$

Putting the N errors together and writing them in a vector form gives,

$$\mathbf{e}_{sp}(\mathbf{r}_s) = \mathbf{A}\boldsymbol{\theta} - \mathbf{b}, \tag{9.24}$$

where,

$$\mathbf{A} \triangleq [\,\mathbf{S} \mid \mathbf{d}\,], \quad \mathbf{S} \triangleq \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ & \vdots & \\ x_N & y_N & z_N \end{bmatrix},$$

$$\boldsymbol{\theta} \triangleq \begin{bmatrix} x_s \\ y_s \\ z_s \\ R_s \end{bmatrix}, \quad \mathbf{b} \triangleq \frac{1}{2}\begin{bmatrix} R_1^2 - d_{10}^2 \\ R_2^2 - d_{20}^2 \\ \vdots \\ R_N^2 - d_{N0}^2 \end{bmatrix},$$

and [**S** I **d**] indicates that **S** and **d** are stacked side-by-side. The corresponding LS criterion is then given by:

$$J_{sp} = \mathbf{e}_{sp}^T\mathbf{e}_{sp} = [\mathbf{A}\boldsymbol{\theta} - \mathbf{b}]^T[\mathbf{A}\boldsymbol{\theta} - \mathbf{b}]. \tag{9.25}$$

In contrast to the hyperbolic error function (9.19), the spherical error function (9.24) is linear in $\mathbf{r}_\mathrm{s}$ given R_s and vice versa. Therefore, the computational complexity to find a solution will *not* dramatically increase as N gets large.

5.2 SPHERICAL INTERSECTION (SX) ESTIMATOR

The SX source location estimator employs the spherical error and solves the problem in two steps [20]. First, we find the least-squares solution for $\mathbf{r}_\mathrm{s}$ in terms of R_s,

$$\mathbf{r}_\mathrm{s} = \mathbf{S}^\dagger(\mathbf{b} - R_\mathrm{s}\mathbf{d}), \tag{9.26}$$

where,

$$\mathbf{S}^\dagger = \left(\mathbf{S}^T\mathbf{S}\right)^{-1}\mathbf{S}^T$$

is the pseudo-inverse of matrix $\mathbf{S}$. Then, substituting (9.26) into the constraint $R_\mathrm{s}^2 = \mathbf{r}_\mathrm{s}^T\mathbf{r}_\mathrm{s}$ yields a quadratic equation as follows

$$R_\mathrm{s}^2 = \left[\mathbf{S}^\dagger(\mathbf{b} - R_\mathrm{s}\mathbf{d})\right]^T \left[\mathbf{S}^\dagger(\mathbf{b} - R_\mathrm{s}\mathbf{d})\right]. \tag{9.27}$$

After expansion, it becomes

$$aR_\mathrm{s}^2 + bR_\mathrm{s} + c = 0, \tag{9.28}$$

where,

$$a = 1 - \|\mathbf{S}^\dagger\mathbf{d}\|^2,\ b = 2\mathbf{b}^T\mathbf{S}^{\dagger T}\mathbf{S}^\dagger\mathbf{d},\ c = -\|\mathbf{S}^\dagger\mathbf{b}\|^2.$$

The valid (real, positive) root is taken as an estimate of the source range R_s and is then substituted into (9.26) to calculate the SX estimate $\hat{\mathbf{r}}_{\mathrm{s,SX}}$ of the source location.

In the SX estimation procedure, the solution of the quadratic equation (9.28) for the source range R_s is required. This solution must be a positive value by all means. If a real positive root is not available, the SX solution does not *exist.* On the contrary, if both of the roots are real and greater than 0, then the SX solution is not *unique.* In both cases, the SX source location estimator fails to produce a reliable estimate, which is not desirable for a real-time implementation.

5.3 SPHERICAL INTERPOLATION (SI) ESTIMATOR

In order to overcome the drawback of the SX algorithm, a spherical interpolation estimator was proposed in [26] which attempts to relax the restriction $R_\mathrm{s} = \|\mathbf{r}_\mathrm{s}\|$ by estimating R_s in the least-squares sense.

To begin, we substitute the least-squares solution (9.26) into the original spherical equation $\mathbf{A}\boldsymbol{\theta} = \mathbf{b}$ to obtain

$$R_\mathrm{s}\mathbf{P}_{\mathbf{S}^\perp}\mathbf{d} = \mathbf{P}_{\mathbf{S}^\perp}\mathbf{b}, \tag{9.29}$$

where,

$$\mathbf{P}_{\mathbf{S}\perp} \stackrel{\triangle}{=} \mathbf{I}_{N\times N} - \mathbf{S}\mathbf{S}^{\dagger}, \tag{9.30}$$

and $\mathbf{I}_{N\times N}$ is an $N \times N$ identity matrix. Matrix $\mathbf{P}_{\mathbf{S}\perp}$ is a projection matrix that projects a vector, when multiplied by the matrix, onto a space that is orthogonal to the column space of $\mathbf{S}$. Such a projection matrix is symmetric (i.e. $\mathbf{P}_{\mathbf{S}\perp} = \mathbf{P}^T_{\mathbf{S}\perp}$) and idempotent (i.e. $\mathbf{P}_{\mathbf{S}\perp} = \mathbf{P}_{\mathbf{S}\perp} \cdot \mathbf{P}_{\mathbf{S}\perp}$). Then the least-squares solution to (9.29) is given by

$$\hat{R}_{\mathrm{s,SI}} = \frac{\mathbf{d}^T\mathbf{P}_{\mathbf{S}\perp}\mathbf{b}}{\mathbf{d}^T\mathbf{P}_{\mathbf{S}\perp}\mathbf{d}}. \tag{9.31}$$

Substituting this solution into (9.26) yields the SI estimate

$$\hat{\mathbf{r}}_{\mathrm{s,SI}} = \mathbf{S}^{\dagger}\left[\mathbf{I}_{N\times N} - \left(\frac{\mathbf{d}\mathbf{d}^T\mathbf{P}_{\mathbf{S}\perp}}{\mathbf{d}^T\mathbf{P}_{\mathbf{S}\perp}\mathbf{d}}\right)\right]\mathbf{b}. \tag{9.32}$$

In practice, the SI estimator performs better, but is computationally a little bit more complex, than the SX estimator.

5.4 LINEAR-CORRECTION LEAST SQUARES ESTIMATOR

Finding the LSE based on the spherical error criterion (9.25) is a linear minimization problem

$$\min_{\boldsymbol{\theta}} \; (\mathbf{A}\boldsymbol{\theta} - \mathbf{b})^T(\mathbf{A}\boldsymbol{\theta} - \mathbf{b}) \tag{9.33}$$

subject to a quadratic constraint

$$\boldsymbol{\theta}^T\boldsymbol{\Sigma}\boldsymbol{\theta} = 0, \tag{9.34}$$

where $\boldsymbol{\Sigma} \stackrel{\triangle}{=} \mathrm{diag}(1, 1, 1, -1)$ is a diagonal and orthonormal matrix.

For such a constrained minimization problem, the technique of Lagrange multipliers will be used and the source location is determined by minimizing the Lagrangian

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}, \lambda) &= J_{\mathrm{sp}} + \lambda\boldsymbol{\theta}^T\boldsymbol{\Sigma}\boldsymbol{\theta} \\ &= (\mathbf{A}\boldsymbol{\theta} - \mathbf{b})^T(\mathbf{A}\boldsymbol{\theta} - \mathbf{b}) + \lambda\boldsymbol{\theta}^T\boldsymbol{\Sigma}\boldsymbol{\theta}, \end{aligned}$$

where λ is the Lagrange multiplier. Expanding this expression yields

$$\mathcal{L}(\boldsymbol{\theta}, \lambda) = \boldsymbol{\theta}^T(\mathbf{A}^T\mathbf{A} + \lambda\boldsymbol{\Sigma})\boldsymbol{\theta} - 2\mathbf{b}^T\mathbf{A}\boldsymbol{\theta} + \mathbf{b}^T\mathbf{b}. \tag{9.35}$$

Necessary conditions for minimizing (9.35) can be obtained by taking the gradient of $\mathcal{L}(\boldsymbol{\theta}, \lambda)$ with respect to $\boldsymbol{\theta}$ and equating the result to zero. This produces:

$$\frac{\partial\mathcal{L}(\boldsymbol{\theta}, \lambda)}{\partial\boldsymbol{\theta}} = 2\left(\mathbf{A}^T\mathbf{A} + \lambda\boldsymbol{\Sigma}\right)\boldsymbol{\theta} - 2\mathbf{A}^T\mathbf{b} = \mathbf{0}. \tag{9.36}$$

Solving for $\boldsymbol{\theta}$ yields the constrained least squares estimate

$$\hat{\boldsymbol{\theta}} = \left(\mathbf{A}^T\mathbf{A} + \lambda\boldsymbol{\Sigma}\right)^{-1}\mathbf{A}^T\mathbf{b}, \tag{9.37}$$

where λ is yet to be determined.

In order to find λ, we can impose the quadratic constraint directly by substituting (9.37) into (9.34), which leads to

$$\mathbf{b}^T\mathbf{A}\left(\mathbf{A}^T\mathbf{A} + \lambda\boldsymbol{\Sigma}\right)^{-1}\boldsymbol{\Sigma}\left(\mathbf{A}^T\mathbf{A} + \lambda\boldsymbol{\Sigma}\right)^{-1}\mathbf{A}^T\mathbf{b} = 0. \tag{9.38}$$

With eigenvalue analysis, the matrix $\mathbf{A}^T\mathbf{A}\boldsymbol{\Sigma}$ can be decomposed as

$$\mathbf{A}^T\mathbf{A}\boldsymbol{\Sigma} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^{-1}, \tag{9.39}$$

where $\boldsymbol{\Lambda} = \text{diag}(\gamma_1, ..., \gamma_4)$ and γ_i, $i = 1, ..., 4$, are the eigenvalues of the matrix $\mathbf{A}^T\mathbf{A}\boldsymbol{\Sigma}$. Substituting (9.39) into (9.38), we may rewrite the constraint as:

$$\mathbf{p}^T(\boldsymbol{\Lambda} + \lambda\mathbf{I})^{-2}\mathbf{q} = 0, \tag{9.40}$$

where

$$\begin{aligned} \mathbf{p} &= \mathbf{U}^T\boldsymbol{\Sigma}\mathbf{A}^T\mathbf{b}, \\ \mathbf{q} &= \mathbf{U}^T\mathbf{A}^T\mathbf{b}. \end{aligned}$$

Define a function of the Lagrange multiplier as follows

$$\begin{aligned} f(\lambda) &\triangleq \mathbf{p}^T(\boldsymbol{\Lambda} + \lambda\mathbf{I})^{-2}\mathbf{q} \\ &= \sum_{i=1}^{4} \frac{p_i q_i}{(\lambda + \gamma_i)^2}. \end{aligned} \tag{9.41}$$

This is a polynomial of degree six and because of its complexity numerical methods need to be used for root searching. Since the root of (9.41) for λ is not unique, a two-step procedure will be followed such that the desired source location could be found.

5.4.1 Unconstrained Spherical Least Squares Estimator. In the first step, we assume that x_s, y_s, z_s, and R_s are mutually independent or equivalently disregard the quadratic constraint (9.34) in purpose. Then the LS solution minimizing (9.25) for $\boldsymbol{\theta}$ (the source location as well as its range) is given by

$$\hat{\boldsymbol{\theta}}_1 = \mathbf{A}^\dagger\mathbf{b}, \tag{9.42}$$

where

$$\mathbf{A}^\dagger = \left(\mathbf{A}^T\mathbf{A}\right)^{-1}\mathbf{A}^T$$

is the pseudo-inverse of the matrix **A**.

A good parameter estimator first and foremost needs to be unbiased. For such an unconstrained spherical least squares estimator, the bias and covariance matrix can be approximated by using the following perturbation analysis method.

When measurement errors are present in the range differences, **A**, **b**, and the parameter estimate $\hat{\boldsymbol{\theta}}_1$ deviate from their true values and can be expressed as:

$$\mathbf{A} = \mathbf{A}^{\mathrm{t}} + \Delta\mathbf{A},\ \mathbf{b} = \mathbf{b}^{\mathrm{t}} + \Delta\mathbf{b},\ \hat{\boldsymbol{\theta}}_1 = \boldsymbol{\theta}^{\mathrm{t}} + \Delta\boldsymbol{\theta}, \tag{9.43}$$

where variables with superscript t denote the true values which also satisfy

$$\boldsymbol{\theta}^{\mathrm{t}} = \mathbf{A}^{\mathrm{t}\dagger}\mathbf{b}^{\mathrm{t}}. \tag{9.44}$$

If the magnitudes of the perturbations are small, the second-order errors are insignificant compared to their first-order counterparts and therefore can be neglected for simplicity, which then yields:

$$\Delta\mathbf{A} = \left[\ \mathbf{0} \mid \boldsymbol{\epsilon}\ \right],\ \ \Delta\mathbf{b} \approx -\mathbf{d}^{\mathrm{t}} \odot \boldsymbol{\epsilon}, \tag{9.45}$$

where $\odot$ denotes the Schur (element-by-element) product. Substituting (9.43) into (9.42) gives,

$$\begin{aligned}&\left(\mathbf{A}^{\mathrm{t}} + \Delta\mathbf{A}\right)^T \left(\mathbf{A}^{\mathrm{t}} + \Delta\mathbf{A}\right)\left(\boldsymbol{\theta}^{\mathrm{t}} + \Delta\boldsymbol{\theta}\right)\\ = \ &\left(\mathbf{A}^{\mathrm{t}} + \Delta\mathbf{A}\right)^T \left(\mathbf{b}^{\mathrm{t}} + \Delta\mathbf{b}\right).\end{aligned} \tag{9.46}$$

Retaining only the linear perturbation terms and using (9.44) and (9.45) produces:

$$\Delta\boldsymbol{\theta} \approx -\mathbf{A}^{\mathrm{t}\dagger}\left(\mathbf{D}\boldsymbol{\epsilon}\right), \tag{9.47}$$

where,

$$\mathbf{D} \triangleq \mathrm{diag}(D_1, D_2, ..., D_N),$$

is a diagonal matrix. Since the measurement error $\boldsymbol{\epsilon}$ in the range differences has zero mean, $\hat{\boldsymbol{\theta}}_1$ is an unbiased estimate of $\boldsymbol{\theta}^{\mathrm{t}}$ when the small error assumption holds:

$$E\{\Delta\boldsymbol{\theta}\} \approx E\left\{-\mathbf{A}^{0\dagger}\mathbf{D}\boldsymbol{\epsilon}\right\} = \mathbf{0}_{4\times 1}. \tag{9.48}$$

The covariance matrix of $\Delta\boldsymbol{\theta}$ is then found as,

$$\mathbf{C}_{\Delta\boldsymbol{\theta}} = E\{\Delta\boldsymbol{\theta}\Delta\boldsymbol{\theta}^T\} = \mathbf{A}^{\mathrm{t}\dagger}\mathbf{D}\mathbf{C}_{\boldsymbol{\epsilon}}\mathbf{D}\mathbf{A}^{\mathrm{t}\dagger^T}, \tag{9.49}$$

where $\mathbf{C}_{\boldsymbol{\epsilon}}$ is known or is properly assumed *a priori.* Theoretically, the covariance matrix $\mathbf{C}_{\Delta\boldsymbol{\theta}}$ cannot be calculated since it contains true values. Nevertheless, it can be approximated by using the values in $\hat{\boldsymbol{\theta}}_1$ with sufficient accuracy, as suggested by our numerical studies.

In the first unconstrained spherical LS estimate (9.42), the range information is redundant because of the independence assumption on the source location and range. If that information is simply discarded, the source location estimate is the same as the SI estimate but with less computational complexity [27]. To demonstrate this, we first write (9.42) into a block form as

$$\hat{\boldsymbol{\theta}}_1 = \begin{bmatrix} \mathbf{S}^T\mathbf{S} & \mathbf{S}^T\mathbf{d} \\ \mathbf{d}^T\mathbf{S} & \mathbf{d}^T\mathbf{d} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{S}^T \\ \mathbf{d}^T \end{bmatrix} \mathbf{b}. \tag{9.50}$$

It can easily be shown that:

$$\begin{bmatrix} \mathbf{S}^T\mathbf{S} & \mathbf{S}^T\mathbf{d} \\ \mathbf{d}^T\mathbf{S} & \mathbf{d}^T\mathbf{d} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{Q} & \mathbf{v} \\ \mathbf{v}^T & k \end{bmatrix}, \tag{9.51}$$

where

$$\begin{aligned} \mathbf{v} &= -\left(\mathbf{S}^T\mathbf{S} - \frac{\mathbf{S}^T\mathbf{d}\mathbf{d}^T\mathbf{S}}{\mathbf{d}^T\mathbf{d}}\right)^{-1} \frac{\mathbf{S}^T\mathbf{d}}{\mathbf{d}^T\mathbf{d}}, \\ \mathbf{Q} &= \left(\mathbf{S}^T\mathbf{S}\right)^{-1}\left[\mathbf{I} - \left(\mathbf{S}^T\mathbf{d}\right)\mathbf{v}^T\right], \\ k &= \frac{1 - (\mathbf{d}^T\mathbf{S})\mathbf{v}}{\mathbf{d}^T\mathbf{d}}. \end{aligned}$$

Next, we define another projection matrix $\mathbf{P}_{\mathbf{d}^\perp}$ associated with the **d**-orthogonal space:

$$\mathbf{P}_{\mathbf{d}^\perp} \triangleq \mathbf{I} - \frac{\mathbf{d}\mathbf{d}^T}{\mathbf{d}^T\mathbf{d}}, \tag{9.52}$$

and find

$$\mathbf{v} = -\left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1} \frac{\mathbf{S}^T\mathbf{d}}{\mathbf{d}^T\mathbf{d}}, \tag{9.53}$$

$$\mathbf{Q} = \left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1}. \tag{9.54}$$

Substituting (9.51) together with (9.53) and (9.54) into (9.50) yields the unconstrained spherical LS estimate for source coordinates,

$$\hat{\mathbf{r}}_{\mathrm{s},1} = \left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1}\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{b}, \tag{9.55}$$

which is the minimizer of

$$J_1(\mathbf{r}_\mathrm{s}) = \|\mathbf{P}_{\mathbf{d}^\perp}\mathbf{b} - \mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\mathbf{r}_\mathrm{s}\|^2, \tag{9.56}$$

or the least-squares solution to the linear equation

$$\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\mathbf{r}_\mathrm{s} = \mathbf{P}_{\mathbf{d}^\perp}\mathbf{b}. \tag{9.57}$$

In fact, the first unconstrained spherical LS estimator tries to approximate the projection of the observation vector $\mathbf{b}$ with the projections of the column vectors of the microphone location matrix $\mathbf{S}$ onto the $\mathbf{d}$-orthogonal space. The source location estimate is the coefficient vector associated with the *best* approximation. Clearly from (9.57), this estimation procedure is the generalization of the plane intersection (PI) method proposed in [19].

By using the Sherman-Morrison formula [28]

$$\left(\mathbf{A}+\mathbf{x}\mathbf{y}^T\right)^{-1}=\mathbf{A}^{-1}-\frac{\mathbf{A}^{-1}\mathbf{x}\mathbf{y}^T\mathbf{A}^{-1}}{1+\mathbf{y}^T\mathbf{A}^{-1}\mathbf{x}}, \tag{9.58}$$

we can expand the item in (9.55) as

$$\left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1}=\left[\mathbf{S}^T\mathbf{S}-\left(\frac{\mathbf{S}^T\mathbf{d}}{\mathbf{d}^T\mathbf{d}}\right)(\mathbf{S}^T\mathbf{d})^T\right]^{-1},$$

and finally can show that the unconstrained spherical LS estimate (9.55) is equivalent to the SI estimate (9.32), i.e. $\hat{\mathbf{r}}_{\mathrm{s},1}\equiv\hat{\mathbf{r}}_{\mathrm{s,SI}}$.

Although the unconstrained spherical LS and the SI estimators are mathematically equivalent, they are quite different in efficiency due to different approaches to the source localization problem. The complexities of the SI and unconstrained spherical LS estimators are in $\mathcal{O}\left(N^3\right)$ and $\mathcal{O}(N)$, respectively. In comparison, the unconstrained spherical LS estimator reduces the complexity of the SI estimator by a factor of N^2, which is significant when N is large (more microphones are used).

5.4.2 Linear Correction. In the previous subsection, we developed the unconstrained spherical LS estimator (USLSE) for source localization and demonstrated that it is mathematically equivalent to the SI estimator but with less computational complexity. Although the USLSE/SI estimates can be accurate as indicated in [27] among others, it is helpful to exploit the redundancy of source range for improving the statistical efficiency (i.e., to reduce the variance of source location estimates) of the overall estimation procedure. Therefore, in the second step, we intend to correct the USLS estimate $\hat{\boldsymbol{\theta}}_1$ to make a better estimate $\hat{\boldsymbol{\theta}}_2$ of $\boldsymbol{\theta}$. This new estimate should be in the neighborhood of $\hat{\boldsymbol{\theta}}_1$ and should obey the constraint (9.34). We expect that the corrected estimate would still be unbiased and would have a smaller variance.

To begin, we substitute $\hat{\boldsymbol{\theta}}_1=\boldsymbol{\theta}^{\mathrm{t}}+\Delta\boldsymbol{\theta}$ into (9.36) and expand the expression to find

$$\mathbf{A}^T\mathbf{A}\hat{\boldsymbol{\theta}}_1+\lambda\boldsymbol{\Sigma}\hat{\boldsymbol{\theta}}_1-(\mathbf{A}^T\mathbf{A}+\lambda\boldsymbol{\Sigma})\Delta\boldsymbol{\theta}=\mathbf{A}^T\mathbf{b}. \tag{9.59}$$

Combined with (9.42), (9.59) becomes

$$(\mathbf{A}^T\mathbf{A}+\lambda\boldsymbol{\Sigma})\Delta\boldsymbol{\theta}=\lambda\boldsymbol{\Sigma}\hat{\boldsymbol{\theta}}_1, \tag{9.60}$$

and hence

$$\Delta\boldsymbol{\theta} = \lambda\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\boldsymbol{\Sigma}\boldsymbol{\theta}^{\mathrm{t}}. \tag{9.61}$$

Substituting (9.61) into $\hat{\boldsymbol{\theta}}_1 = \boldsymbol{\theta}^{\mathrm{t}} + \Delta\boldsymbol{\theta}$ yields

$$\hat{\boldsymbol{\theta}}_1 = \left[\mathbf{I} + \lambda\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\boldsymbol{\Sigma}\right]\boldsymbol{\theta}^{\mathrm{t}}. \tag{9.62}$$

Solving for $\boldsymbol{\theta}^{\mathrm{t}}$ produces the corrected estimate $\hat{\boldsymbol{\theta}}_2$ and also the final output of the linear correction least squares (LCLS) estimator:

$$\hat{\boldsymbol{\theta}}_2 = \left[\mathbf{I} + \lambda\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\boldsymbol{\Sigma}\right]^{-1}\hat{\boldsymbol{\theta}}_1. \tag{9.63}$$

Equation (9.63) suggests how the second-step processing updates the source location estimate based on the first unconstrained spherical least squares result, or equivalently the SI estimate. If the regularity condition [29]

$$\lim_{n\to\infty}\left(\lambda(\mathbf{A}^T\mathbf{A})^{-1}\boldsymbol{\Sigma}\right)^n = \mathbf{0}, \tag{9.64}$$

is satisfied, then the estimate $\hat{\boldsymbol{\theta}}_2$ can be expanded in a Neumann series:

$$\begin{aligned}\hat{\boldsymbol{\theta}}_2 &= \left[\mathbf{I} + \left(-\lambda\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\boldsymbol{\Sigma}\right) + \left(-\lambda\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\boldsymbol{\Sigma}\right)^2 + \cdots\right]\hat{\boldsymbol{\theta}}_1 \\ &= \hat{\boldsymbol{\theta}}_1 + \sum_{n=1}^{\infty}\left[-\lambda\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\boldsymbol{\Sigma}\right]^n\hat{\boldsymbol{\theta}}_1,\end{aligned} \tag{9.65}$$

where the second term is the linear correction. Equation (9.64) implies that in order to avoid divergence, the Lagrange multiplier λ should be small. In addition, λ needs to be determined carefully such that $\hat{\boldsymbol{\theta}}_2$ obeys the quadratic constraint (9.34).

Because the function $f(\lambda)$ is smooth near $\lambda = 0$ (corresponding to the neighborhood of $\hat{\boldsymbol{\theta}}_1$), as suggested by numerical experiments, the secant method [30] can be used to determine its desired root. Two reasonable initial points can be chosen as:

$$\lambda_0 = 0, \quad \lambda_1 = \beta, \tag{9.66}$$

where the small number β is dependent on the array geometry. Five iterations should be sufficient to give an accurate approximation to the root.

The idea of exploiting the relationship between a sound source's range and its location coordinates to improve the estimation efficiency of the SI estimator was first suggested by Chan and Ho in [22] with a quadratic correction. Accordingly, they constructed a quadratic data model for $\hat{\boldsymbol{\theta}}_1$.

$$\hat{\boldsymbol{\theta}}_1 \odot \hat{\boldsymbol{\theta}}_1 = \mathbf{T}(\mathbf{r}_{\mathrm{s}} \odot \mathbf{r}_{\mathrm{s}}) + \mathbf{n}, \tag{9.67}$$

where $\odot$ denotes the Schur (element-by-element) product,

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

is a constant matrix, and $\mathbf{n}$ is the corrupting noise. In contrast to the linear correction technique based on the Lagrange multiplier, the quadratic counterpart needs to know the covariance matrix $\mathbf{C}_{\boldsymbol{\epsilon}}$ of measurement errors in the range differences *a priori*. In a real-time digital source localization system, a poorly estimated $\mathbf{C}_{\boldsymbol{\epsilon}}$ will lead to performance degradation. In addition, the quadratic-correction least squares estimation procedure uses the perturbation approaches to linearly approximate $\Delta\boldsymbol{\theta}$ and $\mathbf{n}$ in (9.43) and (9.67), respectively. Therefore, the approximations of their corresponding covariance matrices $\mathbf{C}_{\Delta\boldsymbol{\theta}}$ and $\mathbf{C}_{\mathbf{n}}$ can be good only when the noise level is low. When noise is at a practically high level, the quadratic-correction least squares estimate has a large bias and a high variance. Furthermore, since the true value of the source location which is necessary for calculating $\mathbf{C}_{\Delta\boldsymbol{\theta}}$ and $\mathbf{C}_{\mathbf{n}}$ cannot be known theoretically, the estimated source location has to be utilized for approximation. It was suggested in [22] that several iterations in the second correction stage would improve estimation accuracy. However, while the bias is suppressed after iterations, the estimate is closer to the SI solution and the variance is boosted, as demonstrated in [24]. Finally, the direct solutions of the quadratic-correction least-squares estimator are the squares of the source location coordinates $(\mathbf{r}_s \odot \mathbf{r}_s)$. In 3-D space, these correspond to 8 positions, which introduce decision ambiguities. Other physical criteria, such as the domain of interest, were suggested but these are hard to define in practical situations, particularly when one of the source coordinates is close to zero.

In comparison, the linear-correction method updates the source location estimate of the first unconstrained spherical LS estimator without making any assumption about the error covariance matrix and without resort to a linear approximation. Even though we need to find a small root of function (9.41) for the Lagrange multiplier λ that satisfies the regularity condition (9.64), the function $f(\lambda)$ is smooth around zero and the solution can be easily determined using the secant method. The linear-correction method achieves a relatively better balance between computational complexity and estimation accuracy.

6. EXAMPLE SYSTEM IMPLEMENTATION

Acoustic source localization systems are not necessarily complicated and need not use computationally powerful and consequently expensive devices for running in real time, as the implementation described briefly in this section demonstrates. The real-time acoustic source localization system with passive microphone arrays for video camera steering in teleconferencing environments was developed by the authors at Bell Laboratories. Figure 9.2 shows a signal-flow diagram of the system.

This system is based on a personal computer that is powered by an Intel Pentium® III 500 MHz general-purposed processor and that runs a Microsoft Windows® operation system. Sonorus AD/24 converter and STUDI/O® digital audio interface card are employed to simultaneously capture multiple microphone signals. The camera is a Sony EVI-D30 with pan, tilt, and zoom capabilities. These motions can be harmoniously performed at the same time by separate motors, providing good coverage of a normal conference room. The host computer drives the camera via two layers of protocols, namely the RS232C serial control protocol and the Video System Control Architecture (VISCA®) protocol. The focus of the video camera is updated four times a second and the video stream is fed into the computer through a video capture card at a rate of 30 frames per second.

The microphone array uses six Lucent Speech Tracker Directional® hypercardioid microphones, as illustrated in Fig. 9.3. The frequency response of these microphones is 200-6000 Hz and beyond 4 kHz there is negligible energy. Therefore microphone signals are sampled at 8 kHz and a one-stage pre-amplifier with the fixed gain 37 dB is used prior to sampling. The reference microphone 0 is located at the center (the origin of the coordinate) and the rest microphones are in the same distance of 40 cm from the reference.

The system incorporates the adaptive eigenvalue decomposition algorithm for time delay estimation and the linear-correction least-squares algorithm for source localization. For comparison, several other time delay estimation and source localization approaches investigated respectively in the previous and current chapter have been also implemented. Subjective testings show that the cutting-edge system is successful and the new acoustic source localization technique is more robust to room reverberation and noise than earlier developed techniques.

7. SOURCE LOCALIZATION EXAMPLES

The empirical bias and standard deviation data in Figs. 9.4 and 9.5 show the results of two source localization examples using four different estimators developed in this chapter (a more comprehensive numerical study can be found in [24]). In the graphs of standard deviation, the CRLBs are also plotted. For the QCLS algorithm, the true value of the source location needs to be known to calculate the covariance matrix of the first-stage SI estimate. But this knowledge is practically inaccessible and the estimated source location has to be used for approximation. It is suggested in [22] that several iterations in the second correction stage could improve the estimation accuracy. In the following, we

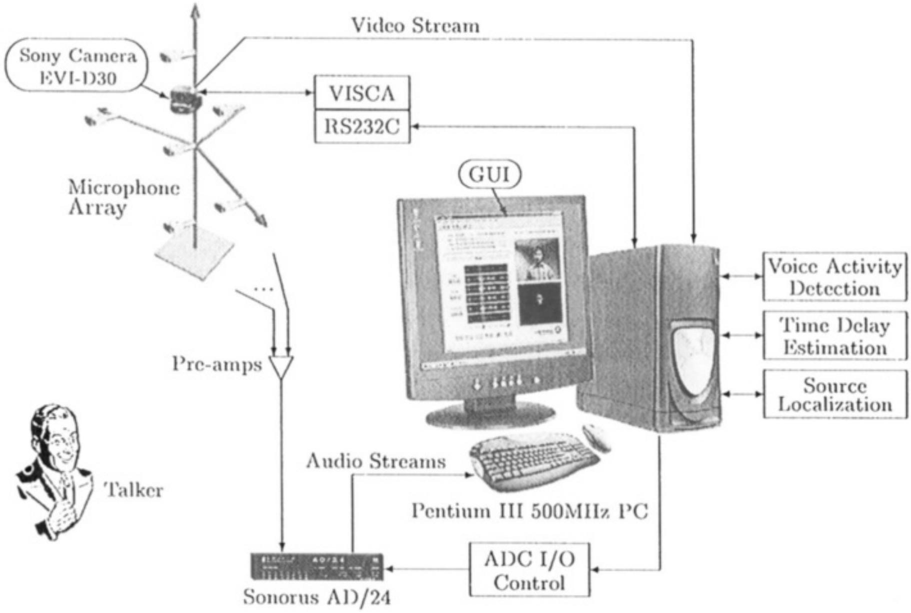


Figure 9.2 Illustration of the real-time acoustic source localization system for video camera steering.

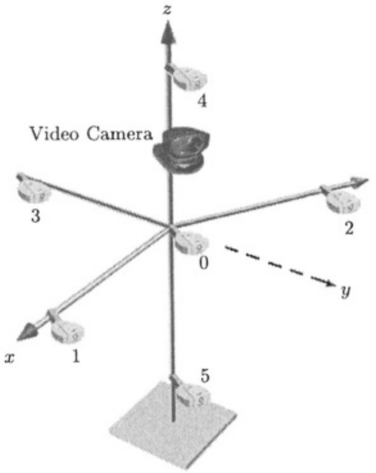


Figure 9.3 Microphone array of the acoustic source localization system for video camera steering.

refer to the one without iterations as the QCLS-i estimator and the other with iterations as the QCLS-ii estimator.

The microphone array designed for the real-time system presented above was used in these examples. As illustrated in Fig. 9.3, the six microphones are located at (distances in centimeters):

$$\begin{array}{lll} \mathbf{r}_0 = (0,0,0), & \mathbf{r}_1 = (40,0,0), & \mathbf{r}_2 = (-40,0,0), \\ \mathbf{r}_3 = (0,0,40), & \mathbf{r}_4 = (0,-40,0), & \mathbf{r}_5 = (0,0,-40). \end{array} \tag{9.68}$$

For such an array, the value of β in (9.66) was empirically set as $\beta = 1$. The source was positioned 300 cm away from the array with a fixed azimuth angle $\theta_s = 45^\circ$ and varying elevation angles γ_s. At each location, the empirical bias and standard deviation of each estimator were obtained by averaging the results of 2000-trial Monte-Carlo runs.

In the first example, errors in time delay estimates are i.i.d. Gaussian with zero mean and 1 cm standard deviation. As seen clearly from Fig. 9.4, the QCLS-i estimator has the largest bias. Performing several iterations in the second stage can effectively reduce the estimation bias, but the solution is more like an SI estimate and the variance is boosted. In terms of standard deviation, all correction estimators perform better than the SI estimator (without correction). Among these four studied LS estimators, the QCLS-ii and the LCLS achieve the lowest standard deviation and their values approach the CRLS at most source locations.

In the second example, measurement errors are mutually dependent and their covariance matrix is given by [22]:

$$\mathbf{C}_\epsilon = \frac{\sigma_\epsilon^2}{2}\begin{bmatrix} 2 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 2 \end{bmatrix}, \tag{9.69}$$

where again $\sigma_\epsilon = 1$ cm. For a more realistic simulation, all estimators are provided with no information of the error distribution. From Fig. 9.5, we see that the performance of each estimator deteriorates because errors are no longer independent. At such a noise level, the linear approximation used by the QCLS estimators is inaccurate and the estimation procedure fails. However, the LCLS estimation procedure makes no assumption about $\mathbf{C}_\epsilon$ and does not depend on a linear approximation. It produces an estimate whose bias and variance are always the smallest.

8. CONCLUSIONS

In this chapter, we have been on a short journey through the fundamental concepts, several cutting-edge estimation algorithms, and some direct applications of acoustic source localization with passive microphone arrays. The localization problem was postulated from a perspective of the estimation theory and the

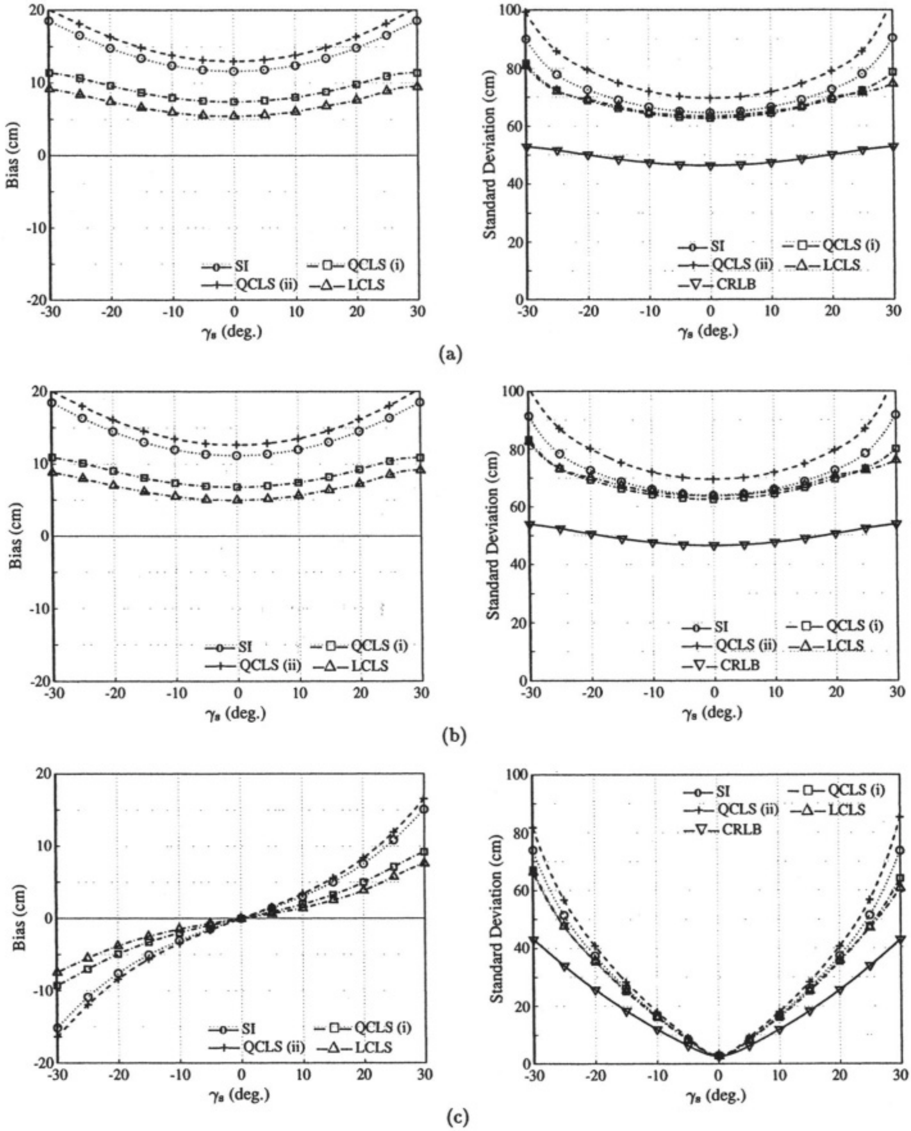


Figure 9.4 Comparisons of empirical bias and standard deviation among the SI, QCLS-i, QCLS-ii, and LCLS estimators with zero mean i.i.d. Gaussian errors of standard deviation $\sigma_\epsilon = 1$ cm. (a) Estimators of x_s, (b) estimators of y_s, (c) estimators of z_s.

Cramèr-Rao lower bound for unbiased location estimators was derived. After an insightful review of conventional approaches ranging from maximum likelihood to least squares estimators, we presented a recently developed linear-correction least-squares algorithm that is more robust to measurement errors

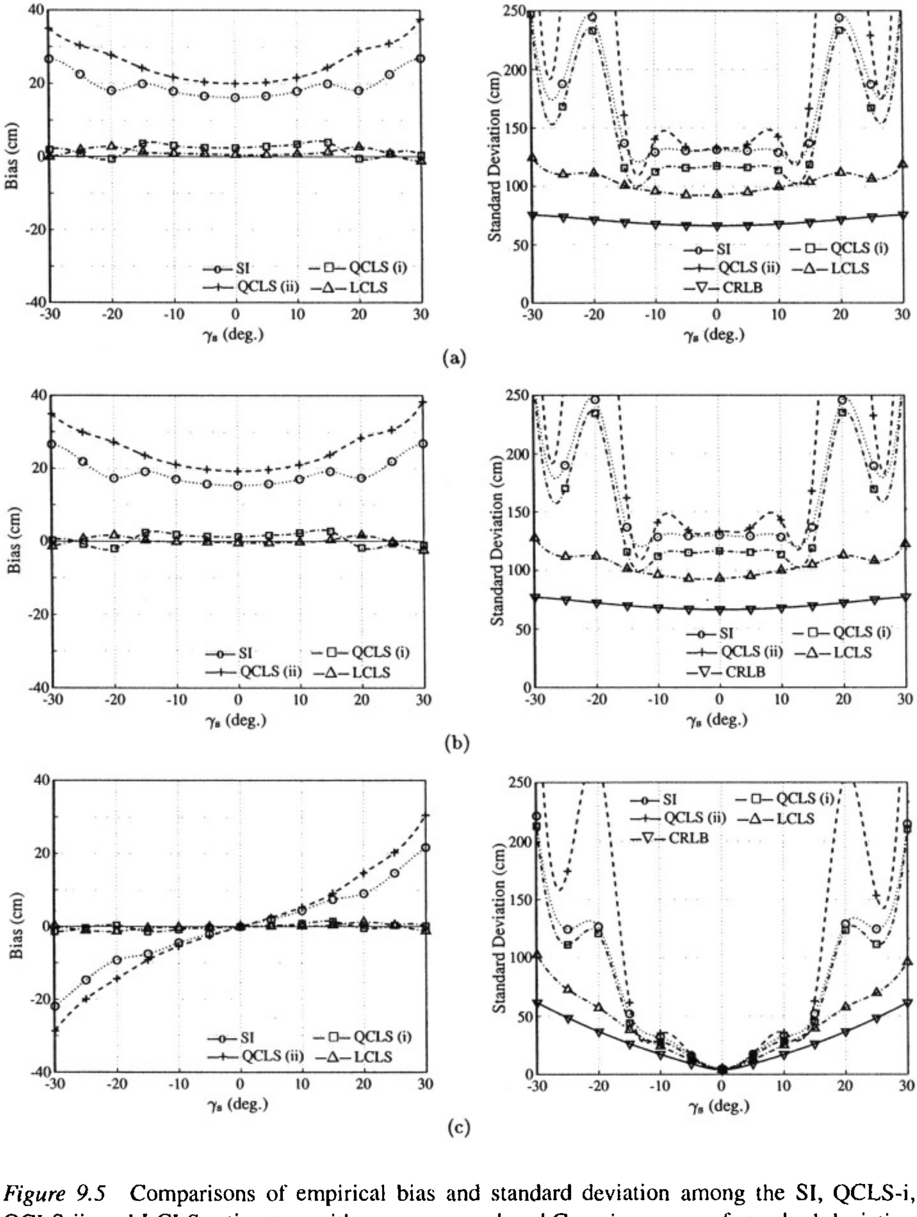


Figure 9.5 Comparisons of empirical bias and standard deviation among the SI, QCLS-i, QCLS-ii, and LCLS estimators with zero mean *colored* Gaussian errors of standard deviation $\sigma_\epsilon = 1$ cm. (a) Estimators of x_s, (b) estimators of y_s, (c) estimators of z_s.

and that is more computationally as well as statistically efficient. Even though very few successful real-time acoustic source localization systems have been developed, to say that implementing such a real-time system needs fast and expensive processors is to drastically overstate its complexity. We presented our acoustic source localization system for video camera steering in teleconferencing, which appealingly used only a cheap Intel Pentium® III general-purposed processor. Acoustic source localization technique will play a significant role in the next-generation multimedia communication systems and a reliable solution will enable the use of many modern array signal processing technologies, most of which assume the knowledge of source locations.

References

[1] S. Haykin, "Radar array processing for angle of arrival estimation," in *Array Signal Processing,* S. Haykin, Ed., Englewood Cliffs, NJ: Prentice-Hall, 1985.

[2] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Magazine,* vol. 13, no. 4, pp. 67–94, July 1996.

[3] R. J. Vaccaro, "The past, present, and future of underwater acoustic signal processing," *IEEE signal Processing Magazine,* vol. 15, pp. 21–51, July 1998.

[4] D. V. Sidorovich and A. B. Gershman, "Two-dimensional wideband interpolated root-MUS1C applied to measured seismic data," *IEEE Trans. Signal Processing,* vol. 46, no. 8, pp. 2263–2267, Aug. 1998.

[5] Y. Huang, J. Benesty, and G. W. Elko, "Microphone arrays for video camera steering," in *Acoustic Signal Processing for Telecommunication,* S. L. Gay and J. Benesty, Eds., Boston, MA: Kluwer Academic, 2000.

[6] H. Wang and P. Chu, "Voice source localization for automatic camera pointing system in videoconferencing," in *Proc. IEEE ASSP Workshop Appls. Signal Processing Audio Acoustics,* 1997.

[7] D. V. Rabinkin, R. J. Ranomeron, J. C. French, and J. L. Flanagan, "A DSP implementation of source location using microphone arrays," in *Proc. SPIE,* vol. 2846, pp. 88–99, 1996.

[8] C. Wang and M. S. Brandstein, "A hybrid real-time face tracking system," in *Proc. IEEE ICASSP,* 1998, vol. 6, pp. 3737–3741.

[9] D. R. Fischell and C. H. Coker, "A speech direction finder," in *Proc. IEEE ICASSP,* 1984, pp. 19.8.1–19.8.4.

[10] H. F. Silverman, "Some analysis of microphone arrays for speech data analysis," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 35, pp. 1699–1712, Dec. 1987.

[11] J. L. Flanagan, A. Surendran, and E. Jan, "Spatially selective sound capture for speech and audio processing," *Speech Communication,* vol. 13, pp. 207–222, Jan. 1993.

[12] D. B. Ward and G. W. Elko, "Mixed nearfield/farfield beamforming: a new technique for speech acquisition in a reverberant environment," in *Proc. IEEE ASSP Workshop Appls. Signal Processing Audio Acoustics,* 1997.

[13] W. R. Hahn and S. A. Tretter, "Optimum processing for delay-vector estimation in passive signal arrays," *IEEE Trans. Inform. Theory,* vol. IT-19, pp. 608–614, May 1973.

[14] M. Wax and T. Kailath, "Optimum localization of multiple sources by passive arrays," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-31, no. 5, pp. 1210–1218, Oct. 1983.

[15] P. E. Stoica and A. Nehorai, "MUSIC, maximum likelihood and Cramèr-Rao bound," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 37, pp. 720–740, May 1989.

[16] J. C. Chen, R. E. Hudson, and K. Yao, "Maximum-likelihood source localization and unknown sensor location estimation for wideband signals in the near-field," *IEEE Trans. Signal Processing,* vol. 50, pp. 1843–1854, Aug. 2002.

[17] Y. Bard, *Nonlinear Parameter Estimation,* New York: Academic Press, 1974.

[18] W. H. Foy, "Position-location solutions by Taylor-series estimation," *IEEE Trans. Aerosp. Electron. Syst.,* vol. AES-12, pp. 187–194, Mar. 1976.

[19] R. O. Schmidt, "A new approach to geometry of range difference location," *IEEE Trans. Aerosp. Electron.,* vol. AES-8, pp. 821–835, Nov. 1972.

[20] H. C. Schau and A. Z. Robinson, "Passive source localization employing intersecting spherical surfaces from time-of-arrival differences," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-35, no. 8, pp. 1223–1225, Aug. 1987.

[21] J. O. Smith and J. S. Abel, "Closed-form least-squares source location estimation from range-difference measurements," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-35, no. 12, pp. 1661–1669, Dec. 1987.

[22] Y. T. Chan and K. C. Ho, "A simple and efficient estimator for hyperbolic location," *IEEE Trans. Signal Processing,* vol. 42, no. 8, pp. 1905–1915, Aug. 1994.

[23] Y. T. Chan and K. C. Ho, "An efficient closed-form localization solution from time difference of arrival measurements," in *Proc. IEEE ICASSP,* 1994, vol. II, pp. 393–396.

[24] Y. Huang, J. Benesty, G. W. Elko, and R. M. Mersereau, "Real-time passive source localization: an unbiased linear-correction least-squares approach," *IEEE Trans. Speech Audio Processing,* vol. 9, no. 8, pp. 943–956, Nov. 2001.

[25] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory,* Englewood Cliffs, New Jersey: Prentice-Hall, 1993.

[26] J. S. Abel and J. O. Smith, "The spherical interpolation method for closed-form passive source localization using range difference measurements echo cancelation," in *Proc. IEEE ICASSP,* 1987, vol. 1, pp. 471–474.

[27] Y. Huang, J. Benesty, and G. W. Elko, "Passive acoustic source localization for video camera steering," in *Proc. IEEE ICASSP,* 2000, vol. 2, pp. 909–912.

[28] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms,* Upper Saddle River, NJ: Prentice-Hall, 1999.

[29] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra,* Philadelphia, PA: SIAM, 2000.

[30] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing,* Cambridge: Cambridge University Press, 1988.

This page intentionally left blank

Chapter 10

BLIND SOURCE SEPARATION FOR CONVOLUTIVE MIXTURES: A UNIFIED TREATMENT

Herbert Buchner
University of Erlangen–Nuremberg
buchner@LNT.de

Robert Aichner
University of Erlangen–Nuremberg
aichner@LNT.de

Walter Kellermann
University of Erlangen–Nuremberg
wk@LNT.de

Abstract Blind source separation (BSS) algorithms for time series can exploit three properties of the source signals: nonwhiteness, nonstationarity, and nongaussianity. While methods utilizing the first two properties are usually based on second-order statistics (SOS), higher-order statistics (HOS) must be considered to exploit nongaussianity. In this chapter, we consider all three properties simultaneously to design BSS algorithms for convolutive mixtures within a new generic framework. This concept derives its generality from an appropriate matrix notation combined with the use of multivariate probability densities for considering the time-dependencies of the source signals. Based on a generalized cost function we rigorously derive the corresponding time-domain and frequency-domain broadband algorithms. Due to the broadband approach, time-domain constraints are obtained which provide a more detailed understanding of the internal permutation problem in traditional narrowband frequency-domain BSS. For both, the time-domain and the frequency-domain versions, we discuss links to well-known and also to novel algorithms that follow as special cases of the framework. Moreover, we use models for correlated spherically invariant random processes (SIRPs) which are well suited for a variety of source signals including speech to obtain efficient solutions in the HOS case. The concept provides a basis for off-line, on-line, and block-on-line algorithms by introducing a general weighting function, thereby allowing for tracking of time-varying real acoustic environments.

Keywords: Blind Source Separation, Convolutive Mixtures, Second-Order Statistics, Higher-Order Statistics, Time Domain, Frequency Domain, Broadband Approach, Spherically Invariant Random Processes

1. INTRODUCTION

The problem of separating convolutive mixtures of unknown time series arises in several application domains, a prominent example being the so-called cocktail party problem, where we want to recover the speech signals of multiple speakers who are simultaneously talking in a room. The room will generally be reverberant due to reflections on the walls, i.e., the original source signals $s_q(n)$, $q = 1,\ldots, Q$ of our separation problem are filtered by a linear multiple input and multiple output (MIMO) system before they are picked up by the sensors. Most commonly used BSS algorithms are developed under the assumption that the number Q of source signals $s_q(n)$ equals the number P of sensor signals $x_p(n)$. However, the more general scenario with an arbitrary number of sources and sensors can always be reduced to the standard BSS model (Fig. 10.1). The case that the sensors outnumber the sources is termed *overdetermined* BSS ($P > Q$). The main approach to simplify the separation problem in this case is to apply principle component analysis (PCA) [1], extract the first P components and then use standard BSS algorithms. The more difficult case $P < Q$ is called *underdetermined* BSS or BSS with *overcomplete bases.* Mostly the sparseness of the sources in the time-frequency domain is used to determine clusters which correspond to the separated sources (e.g., [2]). Recent developments showed that the sparseness can be exploited to eliminate $Q - P$ sources, and then again standard BSS algorithms can be applied [3].

Throughout this chapter, we therefore regard the standard BSS model where the number Q of source signals $s_q(n)$ equals the number of sensor signals $x_p(n)$, $p = 1, \ldots, P$ (Fig. 10.1). An M-tap mixing system is thus described by

$$x_p(n) = \sum_{q=1}^{P} \sum_{\kappa=0}^{M-1} h_{qp}(\kappa) s_q(n - \kappa), \tag{10.1}$$

where $h_{qp}(\kappa)$, $\kappa = 0, \ldots, M - 1$ denote the coefficients of the finite impulse response (FIR) filter model from the q-th source to the p-th sensor.

In BSS, we are interested in finding a corresponding demixing system according to Fig. 10.1, where the output signals $y_q(n)$, $q = 1,\ldots, P$ are described

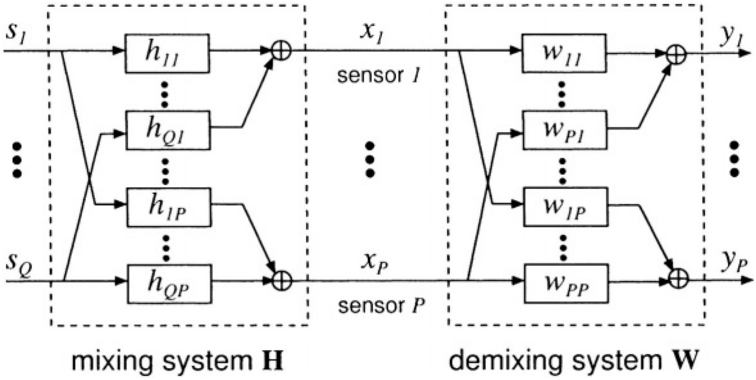


Figure 10.1 Linear MIMO model for BSS.

by

$$y_q(n) = \sum_{p=1}^{P} \sum_{\kappa=0}^{L-1} w_{pq}(\kappa) x_p(n-\kappa). \tag{10.2}$$

The separation of the mixtures obtained by the sensor signals $x_p(n)$ utilizes the fundamental assumption of statistical independence between the original source signals $s_q(n)$. It can be shown (see, e.g., [1]) that the MIMO demixing system coefficients $w_{pq}(\kappa)$ can in fact reconstruct the sources up to an unknown permutation of their order and an unknown filtering of the individual signals, where the demixing filter length L should be chosen at least equal to M. It should be stressed that the filtering ambiguity prevents a deconvolution of the sensor signals and therefore BSS achieves a mere separation of statistically independent signals.

From the description of the BSS model (see Fig. 10.1) it can be seen that this technique is closely related to adaptive beamforming. This relationship was first shown in [4] where BSS was also termed blind beamforming. Thus, as an inherent advantage of BSS, prior knowledge of the spatial position of the sensors and sources is not necessary and, therefore, BSS is robust against unknown array deformations or distortions of the wavefront. Another important difference is the optimization criterion in BSS which utilizes the statistical independence of the source signals. Thus, adaptation of the demixing system is possible even if all source signals are simultaneously active in contrast to adaptive beamforming where the distinction between target signal activity and interfering signal activity has to be made [5]. However, one drawback of most BSS algorithms is that currently the number of sources has to be known for estimating the demixing system.

In [6] it was shown that merely decorrelating the output signals $y_q(n)$ does not lead to a separation of the sources. This implies that we have to force the output signals to become statistically decoupled up tojoint moments of a certain order by using additional conditiones. This can be realized by using approaches to blindly estimate the P^2L MIMO coefficients $w_{pq}(\kappa)$ in (10.2) by exploiting one of the following source signal properties [1]:

(i) Nonwhiteness. Exploited by simultaneous diagonalization of output correlation matrices over multiple time-lags, e.g., [7, 8].

(ii) Nonstationarity. Exploited by simultaneous diagonalization of short-time output correlation matrices at different time instants, e.g., [6], [9]-[17].

(iii) Nongaussianity. Exploited by using higher order statistics for independent component analysis (ICA), e.g., [18]-[23].

While there are several algorithms for convolutive mixtures - both in the time domain and in the frequency domain - utilizing one of these properties, few algorithms explicitly exploit two properties [24, 25] and so far, none is known which simultaneously exploits all three properties. However, it has recently been shown that in practical scenarios, the combination of these criteria can lead to improved performance [24, 25].

Extending the work in [26, 27], we present in the following a rigorous derivation of a unified framework for convolutive mixtures exploiting all three signal properties by using HOS. This is made possible by introducing an appropriate matrix notation combined with the use of multivariate probability densities for considering the time-dependencies of the source signals. The approach is suitable for on-line and off-line algorithms as it uses a general weighting function, thereby allowing for tracking of time-varying environments [28]. The processing delay can be kept low by working with overlapping and/or partitioned signal blocks [29]. Having derived a generic time-domain algorithm, we introduce a model for spherically invariant random processes (SIRPs) [30] which are well suited, e.g., for speech to allow efficient realizations. Moreover, we discuss links to well-known SOS algorithms and we show that a previously presented algorithm [26] is the optimum second-order BSS approach in the sense of minimum mutual information known from information theory. Furthermore we introduce an equivalent broadband formulation in the frequency domain by extending the tools of [31] to unsupervised adaptive filtering. This will also give a detailed insight in the internal permutation problem of narrowband frequency-domain BSS. Again, links to well-known and extended HOS and SOS algorithms as special cases are discussed. Moreover, using the so-called generalized coherence [32], links between the time-domain and frequency-domain SOS algorithms can be established [26] showing that our cost function leads to an update equation with an inherent normalization. As shown by experimental results, this allows an efficient separation of real-world speech signals.

2. GENERIC BLOCK TIME-DOMAIN BSS ALGORITHM

In this section, we first introduce a general matrix formulation as a basis for a rigorous derivation of time-domain algorithms from a cost function which inherently takes into account all three fundamental signal properties (i)-(iii). We then consider the so-called equivariance property in the convolutive case for deriving the corresponding natural gradient update. From this formulation, several well-known and novel algorithms follow as special cases.

2.1 MATRIX NOTATION FOR CONVOLUTIVE MIXTURES

From Fig. 10.1, it can be seen that the output signals $y_q(n)$ are obtained by convolving the input signals $x_p(n)$ with the demixing filter coefficients w_{pq}. In addition to the filter length L and the number of channels P we need to introduce two more parameters for the following general formulation:

- the number of time-lags D taken into account for exploiting the non-whiteness property of the input signals as shown below ($1 \leq D \leq L$), and
- the block length N as basis for averaging the estimates of the multivariate probability density functions (pdfs) as used below ($N > PD$ in general; $N > D$ for the natural gradient update discussed below).

To derive an algorithm for block processing of convolutive mixtures taking into account D time-lags, we first need to reformulate the convolution (10.2):

$$\mathbf{y}_q(m,j) = \sum_{p=1}^{P} \mathbf{x}_p(m,j)\mathbf{W}_{pq}, \tag{10.3}$$

where m denotes the block index, and $j = 0, \cdots, N-1$ is a time-shift index within a block of length N, and

$$\begin{aligned} \mathbf{x}_p(m,j) &= [x_p(mL+j), \ldots, x_p(mL-2L+1+j)], & (10.4)\\ \mathbf{y}_q(m,j) &= [y_q(mL+j), \ldots, y_q(mL-D+1+j)]. & (10.5) \end{aligned}$$

The $2L \times D$ matrix $\mathbf{W}_{pq}$ exhibits a Sylvester structure that contains all L coefficients of the respective demixing filter in each column needed for the matrix formulation of the linear convolution:

$$\mathbf{W}_{pq} = \begin{bmatrix} w_{pq,0} & 0 & \cdots & 0 \\ w_{pq,1} & w_{pq,0} & \ddots & \vdots \\ \vdots & w_{pq,1} & \ddots & 0 \\ w_{pq,L-1} & \vdots & \ddots & w_{pq,0} \\ 0 & w_{pq,L-1} & \ddots & w_{pq,1} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & w_{pq,L-1} \\ 0 & \cdots & 0 & 0 \\ \vdots & \cdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix}. \tag{10.6}$$

It can be seen that for the general case, $1 \leq D \leq L$, the last $L - D + 1$ rows are padded with zeros to ensure compatibility with the length of $\mathbf{x}_p(m, j)$ with regard to a concise frequency-domain formulation in Sect. 3. Finally, to allow a convenient notation of the algorithm we combine all channels, and thus we can write (10.3) compactly as

$$\mathbf{y}(m, j) = \mathbf{x}(m, j)\mathbf{W}, \tag{10.7}$$

with

$$\mathbf{x}(m, j) = [\mathbf{x}_1(m, j), \ldots, \mathbf{x}_P(m, j)], \tag{10.8}$$

$$\mathbf{y}(m, j) = [\mathbf{y}_1(m, j), \ldots, \mathbf{y}_P(m, j)], \tag{10.9}$$

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{11} & \cdots & \mathbf{W}_{1P} \\ \vdots & \ddots & \vdots \\ \mathbf{W}_{P1} & \cdots & \mathbf{W}_{PP} \end{bmatrix}. \tag{10.10}$$

Also, with respect to the frequency-domain derivation in Sect. 3. we extend (10.7) by collecting all N vectors $\mathbf{x}_p$, $\mathbf{y}_q$, so that all output signal samples of the m-th block are captured:

$$\mathbf{Y}(m) = \mathbf{X}(m)\mathbf{W}, \tag{10.11}$$

with the matrices

$$\mathbf{Y}(m) = [\mathbf{Y}_1(m), \cdots, \mathbf{Y}_P(m)], \tag{10.12}$$

$$\mathbf{X}(m) = [\mathbf{X}_1(m), \cdots, \mathbf{X}_P(m)], \tag{10.13}$$

$$\mathbf{Y}_q(m) = [\mathbf{y}_q^T(m, 0), \ldots, \mathbf{y}_q^T(m, N-1)]^T, \tag{10.14}$$

$$\mathbf{X}_p(m) = [\mathbf{x}_p^T(m, 0), \ldots, \mathbf{x}_p^T(m, N-1)]^T. \tag{10.15}$$

Superscript ^{T} denotes the transposition of a vector or a matrix. Obviously, $\mathbf{X}_p(m), p = 1, \ldots, P$ in (10.15) are Toeplitz matrices of size ($N \times 2L$) due to the shift of subsequent rows by one sample each:

$$\mathbf{X}_p(m) = \begin{bmatrix} x_p(mL) & \cdots & x_p(mL-2L+1) \\ x_p(mL+1) & \ddots & x_p(mL-2L+2) \\ \vdots & \ddots & \vdots \\ x_p(mL+N-1) & \cdots & x_p(mL-2L+N) \end{bmatrix}. \qquad (10.16)$$

Analogously to supervised block-based adaptive filtering [29, 31], the approach followed here can also be carried out with overlapping and/or partitioned data blocks to increase the convergence rate and to reduce the signal delay. Overlapping is done by simply replacing the time index mL in the equations by $m\frac{L}{\alpha}$ with the overlap factor $1 \leq \alpha \leq L$. For clarity, we will omit the overlap factor and will point to it when necessary.

2.2 COST FUNCTION AND ALGORITHM DERIVATION

A generic SOS algorithm for convolutive mixtures has been derived in [26] from a cost function that explicitly contains correlation matrices that include several time-lags (c.f. property (i)) under the assumption of short-time stationarity (c.f. property (ii)). Additionally, for exploiting property (iii), higher order statistics have to be considered. Higher-order approaches for BSS can be divided into three classes [1]: maximum likelihood (ML) estimation [21], minimization of the mutual information (MMI) among the output signals [33], and maximization of the entropy (ME/'infomax') [23]. Although all of these HOS approaches lead to similar update rules, MMI can be regarded as the most general one [33].

Based on a generalization of Shannon's mutual information [34], we now define the following cost function which simultaneously accounts for the three fundamental properties (i)-(iii):

$$\begin{aligned} \mathcal{J}(m) &= -\sum_{i=0}^{\infty} \beta(i,m) \frac{1}{N} \sum_{j=0}^{N-1} \{\log(\hat{p}_{1,D}(\mathbf{y}_1(i,j)) \cdot \ldots \cdot \hat{p}_{P,D}(\mathbf{y}_P(i,j))) \\ &\quad - \log(\hat{p}_{PD}(\mathbf{y}_1(i,j), \ldots, \mathbf{y}_P(i,j)))\}, \qquad (10.17) \end{aligned}$$

where $\hat{p}_{p,D}(\cdot)$ is the estimated or assumed *multivariate* probability density function (pdf) for channel p of dimension D and $\hat{p}_{PD}(\cdot)$ is the joint pdf of dimension PD over all channels. Furthermore, D is the memory length, i.e., the number of time-lags to model the nonwhiteness of the P signals as above. Note also that the time series of these pdf estimates completely describes any multichannel stochastic process with the assumption of short-time stationarity over length-N blocks (this assumption is reasonable for many real-world signals such as speech). The expectation operator of the mutual information [34] is replaced in (10.17) by short-time averages within these blocks. β is a window function with finite support that is normalized according to $\sum_{i=0}^{\infty} \beta(i,m) = 1$, and allows off-line, on-line, and block-online implementations of the algorithms. As an example, $\beta(i,m) = (1-\lambda)\lambda^{m-i}$ for $0 \leq i \leq m$, and $\beta(i,m) = 0$ else, leads to an efficient on-line version allowing for tracking in time-varying environments [28].

In this chapter, we consider algorithms based on first-order gradients. An extension to higher-order gradients would be straightforward but computationally more expensive. Moreover, to obtain general expressions allowing a smooth transition to the frequency domain, we consider complex signals for the derivative. In order to calculate the gradient [35, 36]

$$\nabla_{\mathbf{W}} \mathcal{J}(m) = 2\frac{\partial \mathcal{J}(m)}{\partial \mathbf{W}^*}, \tag{10.18}$$

we need to express the cost function (10.17) in terms of the demixing matrix **W** which contains the coefficients of all channels. A common way to achieve this is to transform the output signal pdf $\hat{p}_{PD}(\mathbf{y})$ into the *PD*-dimensional input signal pdf using **W** which is considered as a mapping matrix for this linear transformation. This procedure is directly applied to the second term in the braces of (10.17), followed by differentiation w.r.t. **W**. The derivative of the input signal pdf, which appears as an additive constant due to the logarithm, vanishes as it is independent of **W**. The argument of the logarithm in the first term in the braces, however, is factorized among the channels. Therefore, we apply the chain rule in this case, rather than transforming the pdfs.

Finally, the generic HOS gradient for the coefficient update utilizing all three signal properties (i)-(iii) can be expressed as

$$\begin{aligned}\nabla_{\mathbf{W}} \mathcal{J}(m) &= \frac{2}{N}\sum_{i=0}^{\infty} \beta(i,m) \sum_{j=0}^{N-1} \Big\{ \mathbf{x}^H(i,j)\boldsymbol{\Phi}(\mathbf{y}(i,j)) \\ &\quad - \mathbf{V}^{1_D 0}_{2LP\times DP}\left(\mathbf{W}^H \mathbf{V}^{1_D 0}_{2LP\times DP}\right)^{-1}\Big\},\end{aligned} \tag{10.19}$$

with the *multivariate score function*

$$\boldsymbol{\Phi}(\mathbf{y}(i,j)) = \left[-\frac{\frac{\partial \hat{p}_{1,D}(\mathbf{y}_1(i,j))}{\partial \mathbf{y}_1(i,j)}}{\hat{p}_{1,D}(\mathbf{y}_1(i,j))}, \ldots, -\frac{\frac{\partial \hat{p}_{P,D}(\mathbf{y}_P(i,j))}{\partial \mathbf{y}_P(i,j)}}{\hat{p}_{P,D}(\mathbf{y}_P(i,j))} \right], \tag{10.20}$$

and the $2LP \times DP$ window matrix $\mathbf{V}^{1_D0}_{2LP\times DP}$ defined as

$$\mathbf{V}^{1_D0}_{2LP\times DP} = \text{bdiag}\left\{\mathbf{W}^{1_D0}_{2L\times D}, \ldots, \mathbf{W}^{1_D0}_{2L\times D}\right\}, \tag{10.21}$$

$$\mathbf{W}^{1_D0}_{2L\times D} = \left[\mathbf{I}_{D\times D}, \mathbf{0}_{(2L-D)\times D}\right]. \tag{10.22}$$

The operator $\textbf{bdiag}\{\mathbf{A}_1, \ldots, \mathbf{A}_P\}$ denotes a block-diagonal matrix with submatrices $\mathbf{A}_1, \ldots, \mathbf{A}_P$ on its diagonal. For the description of window matrices (also appearing in the frequency-domain algorithms in Sect. 3.) we use the following conventions:

- The lower index of a matrix denotes its dimensions.
- *P*-channel matrices (as indicated by the size in the lower index) are partitioned into *P* single-channel window matrices.
- The upper index describes the positions of ones and zeros. Unity submatrices are always located at the upper left ('10') or lower right ('01') corners of the respective single-channel window matrix. The size of these clusters is indicated in subscript (e.g., '$\mathbf{0 1}_L$').

The window matrix $\mathbf{V}^{1_D0}_{2LP\times DP}$ appears due to the transformation of pdfs by the non-square Sylvester matrix $\mathbf{W}$ [37].

With an iterative optimization procedure, the current demixing matrix is obtained by the recursive update equation

$$\mathbf{W}(m) = \mathbf{W}(m-1) - \mu\Delta\mathbf{W}(m), \tag{10.23}$$

where μ is a stepsize parameter, and $\Delta\mathbf{W}(m)$ is the update which is set equal to $\nabla_{\mathbf{W}}\mathcal{J}(m)$ for gradient descent adaptation. Due to the adaptation process, the coefficient matrix becomes time-variant. For clarity we will generally omit the block index of $\mathbf{W}$ and will point to it when necessary. Note that the Sylvester structure (see Eqs. 10.6, 10.10) of the update in (10.23) has to be ensured. (The structure of the update might be disturbed by imprecision effects and also (depends on the technique used for estimating the pdfs.) A simple remedy for this generic update is to pick the first column and replicate it. For special cases, and frequency-domain versions discussed later, we will give more specific solutions for enforcing this constraint.

2.3 EQUIVARIANCE PROPERTY AND NATURAL GRADIENT

It is known that stochastic gradient descent, i.e., $\Delta\mathbf{W}(m) = \nabla_{\mathbf{W}}\mathcal{J}(m)$ suffers from slow convergence in many practical problems due to statistical dependencies in the data being processed.

In the BSS application, we can show that the separation performance of the gradient update rule (10.19), (10.23) depends on the MIMO mixing system. The mixing process can be described analogously to (10.7) by $\mathbf{x}(m,j) = \mathbf{s}(m,j)\mathbf{H}$, where $\mathbf{s}(m,j)$ is the corresponding $1 \times P(M + L{-}1)$ source signal row vector and $\mathbf{H}$ is the $P(M + \mathrm{L} - 1) \times 2PL$ mixing matrix in Sylvester structure. The dimensions result from the linearity condition of the convolution. Due to the inevitable filtering ambiguity in convolutive BSS (e.g., [1]), it is at best possible to obtain an arbitrary *block diagonal* matrix $\mathbf{C} = \mathbf{HW}$, i.e., $\mathbf{C} - \text{bdiag}\,\mathbf{C} = \mathbf{0}$, where $\mathbf{C}$ combines mixing and unmixing coefficient matrices. This means the output signals can become mutually independent but the output signals are still arbitrarily filtered versions of the source signals. To see how (10.19) behaves, we pre-multiply both sides of (10.19) by $\mathbf{H}$. This way it can easily be shown that $\mathbf{C}(m)$ depends on the mixing system $\mathbf{H}$, and, therefore, on its conditioning.

Fortunately, a modification of the ordinary gradient, termed the *natural gradient* by Amari [20] and the *relative gradient* by Cardoso [21] (which is equivalent to the natural gradient in the BSS application) has been developed that largely removes all effects of an ill-conditioned mixing matrix $\mathbf{H}$ assuming an appropriate initialization of $\mathbf{W}$. The idea of the relative gradient is based on the equivariance property. Generally speaking, an estimator behaves equivariantly if it produces estimates that, under data transformation, are similarly transformed. A key property of equivariant estimators is that they exhibit uniform performance. In [26] the natural/relative gradient has been extended to the case of Sylvester matrices yielding

$$\nabla_{\mathbf{W}}^{\mathrm{NG}}\mathcal{J} = \mathbf{W}\mathbf{W}^{H}\nabla_{\mathbf{W}}\mathcal{J}. \tag{10.24}$$

Together with (10.19) this immediately leads to the following expression:

$$\nabla_{\mathbf{W}}^{\mathrm{NG}}\mathcal{J}(m) = \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\sum_{j=0}^{N-1}\mathbf{W}\left\{\mathbf{y}^{H}(i,j)\mathbf{\Phi}(\mathbf{y}(i,j)) - \mathbf{I}\right\}, \tag{10.25}$$

which is then used as update $\Delta\mathbf{W}$ in (10.23).

In the derivation of the natural gradient for instantaneous mixtures, the fact that the demixing matrices form a so-called Lie group has played an important role [20]. However, the block-Sylvester matrices $\mathbf{W}$ after (10.6), (10.10) do not form a Lie group (as they are generally not invertible). To see that the above formulation of the natural gradient is indeed justified, we again pre-multiply the update (10.25) with $\mathbf{H}$, which leads to

$$\Delta\mathbf{C}(m) = \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\sum_{j=0}^{N-1}\mathbf{C}(i)\left\{\mathbf{y}^{H}(i,j)\mathbf{\Phi}(\mathbf{y}(i,j)) - \mathbf{I}\right\}.$$

Thus, the temporal evolution of $\mathbf{C} = \mathbf{C}(m)$ depends only on the estimated source signal vector sequence and the stepsize μ, and the dependency on the mixing matrix $\mathbf{H}$ has been absorbed as an initial condition into $\mathbf{C}(0) = \mathbf{HW}(0)$ leading to the desired uniform performance of (10.25) proving the equivariance property of the natural gradient.

Another well-known advantage of using the natural gradient is a reduction of the computational complexity of the update as the inversion of the $PD \times PD$ matrix $\mathbf{W}^H\mathbf{V}^{1_D 0}_{2LP\times PD}$ in (10.19) need not be carried out in (10.25). Furthermore, it can be shown for specific pdfs (Sect. 2.4) that instead of $N > PD$ the condition $N > D$ is sufficient for the natural gradient update due to the smaller matrices to be inverted [26].

Moreover, noting that the products of Sylvester matrices $\mathbf{W}_{pq}$ and the remaining matrices in the update equation (10.25) can be described by linear convolutions, they can be efficiently implemented by a fast convolution.

The update in (10.25) represents a so-called holonomic algorithm as it imposes the constraint $\mathbf{y}^H(i,j)\mathbf{\Phi}(\mathbf{y}(i,j)) = \mathbf{I}$ on the magnitudes of the recovered signals. However, when the source signals are nonstationary, these constraints may force a rapid change in the magnitude of the demixing matrix leading to numerical instabilities in some cases (see, e.g., [19]). Replacing I in (10.25) by the term $\mathrm{bdiag}\{\mathbf{y}^H(i,j)\mathbf{\Phi}(\mathbf{y}(i,j))\}$ yields the nonholonomic natural gradient algorithm with improved convergence characteristics for nonstationary sources:

$$\begin{aligned}\nabla^{\mathrm{NG}}_{\mathbf{W}}\mathcal{J}(m) &= \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\sum_{j=0}^{N-1}\mathbf{W}\left\{\mathbf{y}^H(i,j)\mathbf{\Phi}(\mathbf{y}(i,j))\right.\\ &\quad \left.-\mathrm{bdiag}\{\mathbf{y}^H(i,j)\mathbf{\Phi}(\mathbf{y}(i,j))\}\right\}. \qquad (10.26)\end{aligned}$$

Here, the bdiag operator sets all channel-wise cross-terms to zero. Note that the nonholonomic property can also be directly taken into account in the cost function as shown in [27].

2.4 SPECIAL CASES AND LINKS TO KNOWN TIME-DOMAIN ALGORITHMS

The update rules (10.19) and (10.25) provide a very general basis for BSS of convolutive mixtures. However, to apply them in a real-world scenario, an appropriate multivariate score function (10.20) has to be determined, i.e., we have to handle P high-dimensional multivariate pdfs $\hat{p}_{p,D}(\mathbf{y}_p(i,j))$, $p = 1,\ldots,P$. In general, this is a very challenging task, as it includes all corresponding higher-order cumulants (including time-lags which may be on the order of several hundred in real acoustic environments).

In the following we will present an efficient solution for these problems by assuming so-called spherically invariant random processes (SIRPs). Moreover we will show some links to SOS algorithms. Without loss of generality we consider now the case $P = Q = 2$ for simplicity.

2.4.1 Incorporating Spherically Invariant Random Processes (SIRPs) as Signal Model. The SIRP models are representative for a wide class of stochastic processes. It has been shown that speech signals in particular can very accurately be represented by SIRPs [30]. One of the great advantages arising from the SIRP model is that multivariate pdfs can be derived analytically from the corresponding univariate probability density function together with the correlation matrices including time-lags. The correlation matrices can be estimated from the data while for the univariate pdf, we can assume one of the well-known functions for speech signals, e.g., the Laplacian density, or we can estimate the univariate pdf as well, based on parameterized representations, such as the Gram-Charlier or Edgeworth expansions [18].

The general model of a correlated SIRP of D-th order for channel p is given with a properly chosen function $f_{p,D}(\cdot)$ by [30]

$$\hat{p}_{p,D}(\mathbf{y}_p(i,j)) = \frac{1}{\sqrt{\pi^D \det(\mathbf{R}_{\mathbf{y}_p\mathbf{y}_p}(i))}} f_{p,D}\left(\mathbf{y}_p(i,j)\mathbf{R}^{-1}_{\mathbf{y}_p\mathbf{y}_p}(i)\mathbf{y}_p^H(i,j)\right) \tag{10.27}$$

with the $D \times D$ correlation matrix $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_q}$ defined as

$$\mathbf{R}_{\mathbf{y}_p\mathbf{y}_q}(i) = \frac{1}{N}\sum_{j=0}^{N-1} \mathbf{y}_p^H(i,j)\mathbf{y}_q(i,j) = \frac{1}{N}\mathbf{Y}_p^H(i)\mathbf{Y}_q(i). \tag{10.28}$$

As the best known example, the multivariate Gaussian can be viewed as a special case of the class of SIRPs. To calculate the score function for SIRPs in general, we employ the chain rule [36] to Eq. 10.27

$$\frac{\frac{\partial \hat{p}_{p,D}(\mathbf{y}_p(i,j))}{\partial \mathbf{y}_p(i,j)}}{\hat{p}_{p,D}(\mathbf{y}_p(i,j))} = \underbrace{\left[-\frac{1}{f_{p,D}(u_p)}\frac{\partial f_{p,D}(u_p)}{\partial u_p}\right]}_{:=\phi_{p,D}(u_p)} \mathbf{y}_p(i,j)\mathbf{R}^{-1}_{\mathbf{y}_p\mathbf{y}_p}(i), \tag{10.29}$$

where $u_p = \mathbf{y}_p\mathbf{R}^{-1}_{\mathbf{y}_p\mathbf{y}_p}\mathbf{y}_p^H$. For convenience, we call the scalar function $\phi_{p,D}(u_p)$ the *SIRP score* of channel p.

Having derived the multivariate score function for the SIRP model (10.29), we can now introduce it into the generic HOS natural gradient update equation (10.25) with its nonholonomic extension. In the 2-by-2 case, this leads to the following expression for the *nonholonomic* HOS-SIRP update:

$$\Delta\mathbf{W}(m) = 2\sum_{i=0}^{\infty}\beta(i,m)\mathbf{W}\begin{bmatrix} \mathbf{0} & \tilde{\mathbf{R}}_{\mathbf{y}_1\mathbf{y}_2}(i)\mathbf{R}^{-1}_{\mathbf{y}_2\mathbf{y}_2}(i) \\ \tilde{\mathbf{R}}_{\mathbf{y}_2\mathbf{y}_1}(i)\mathbf{R}^{-1}_{\mathbf{y}_1\mathbf{y}_1}(i) & \mathbf{0} \end{bmatrix}, \tag{10.30}$$

where the modified matrices $\tilde{\mathbf{R}}_{\mathbf{y}_p\mathbf{y}_q}$, $p \neq q$ are given by

$$\tilde{\mathbf{R}}_{\mathbf{y}_p\mathbf{y}_q}(i) = \frac{1}{N}\sum_{j=0}^{N-1} \phi_{q,D}\left(\mathbf{y}_q(i,j)\mathbf{R}^{-1}_{\mathbf{y}_q\mathbf{y}_q}(i)\mathbf{y}_q^H(i,j)\right) \mathbf{y}_p^H(i,j)\mathbf{y}_q(i,j), \tag{10.31}$$

$$\phi_{q,D}(u_q) = -\frac{f'_{q,D}(u_q)}{f_{q,D}(u_q)}. \tag{10.32}$$

The SIRP score $\phi_{q,D}(u_q)$ of channel q in (10.31) is a scalar value function which causes a weighting of the correlation matrix.

From the update equation (10.30), we see that the SIRP model leads to an inherent normalization by the auto-correlation submatrices.

To derive a HOS-SIRP realization using (10.32) we need an analytical expression of the multivariate pdfs (10.27) for all channels. As noted above, for SIRPs, these expressions can actually be derived from the univariate pdfs [30]. Following the procedure in [30], we obtain, e.g., as the *optimum SIRP score for univariate Laplacian pdfs* [27]:

$$\phi_{q,D}(u_q) = -\frac{1}{D - \sqrt{2s}\frac{K_{D/2+1}(\sqrt{2u_q})}{K_{D/2}(\sqrt{2u_q})}}, \tag{10.33}$$

where $K_\nu(\cdot)$ denotes the ν-th order modified Bessel function of the second kind.

2.4.2 Generic BSS Based on Second-Order Statistics. To see the link to second-order BSS algorithms we use the model of multivariate Gaussian pdfs in the general cost function (10.17). As for Gaussian pdfs the cost function reduces to SOS we only utilize the nonstationarity and the nonwhiteness of the source signals. We now insert the multivariate Gaussian pdf

$$\hat{p}_{p,D}(\mathbf{y}_p(i,j)) = \frac{1}{\sqrt{(2\pi)^D \det(\mathbf{R}_{\mathbf{y}_p\mathbf{y}_p}(i))}} e^{-\frac{1}{2}\mathbf{y}_p(i,j)\mathbf{R}^{-1}_{\mathbf{y}_p\mathbf{y}_p}(i)\mathbf{y}_p^H(i,j)} \tag{10.34}$$

in the natural gradient update equation of the generic HOS BSS algorithm (10.25). Note that there are several different representations of real and complex Gaussian multivariate pdfs in the literature [37, 38]. The most important ones in practice being the real case for speech and audio applications, and the rotation-invariant complex case mostly used in communication theory. In both cases the elements of the score function $\mathbf{\Phi}(\mathbf{y}(i,j))$ for a Gaussian pdf reduce to

$$\frac{\frac{\partial \hat{p}_{p,D}(\mathbf{y}_p(i,j))}{\partial \mathbf{y}_p(i,j)}}{\hat{p}_{p,D}(\mathbf{y}_p(i,j))} = \mathbf{y}_p(i,j)\mathbf{R}^{-1}_{\mathbf{y}_p\mathbf{y}_p}(i). \tag{10.35}$$

With (10.25) and (10.35) we finally obtain the natural gradient update of the generic SOS BSS algorithm originally introduced in [26]

$$\nabla_{\mathbf{W}}^{\mathrm{NG}} \mathcal{J}(m) = 2\sum_{i=0}^{\infty} \beta(i,m)\mathbf{W}\left\{\mathbf{R}_{\mathbf{yy}}(i) - \operatorname{bdiag}\mathbf{R}_{\mathbf{yy}}(i)\right\}\operatorname{bdiag}^{-1}\mathbf{R}_{\mathbf{yy}}(i) \tag{10.36}$$

with the *PD* × *PD* short-time correlation matrix $\mathbf{R}_{\mathbf{yy}}(i)$ defined as

$$\mathbf{R}_{\mathbf{yy}}(i) = \frac{1}{N}\sum_{j=0}^{N-1} \mathbf{y}^{H}(i,j)\mathbf{y}(i,j) = \frac{1}{N}\mathbf{Y}^{H}(i)\mathbf{Y}(i). \tag{10.37}$$

For the 2 × 2 case we can express (10.36) as

$$\nabla_{\mathbf{W}}^{\mathrm{NG}} \mathcal{J}(m) = 2\sum_{i=0}^{\infty} \beta(i,m)\mathbf{W}\begin{bmatrix} \mathbf{0} & \mathbf{R}_{\mathbf{y}_1\mathbf{y}_2}(i)\mathbf{R}_{\mathbf{y}_2\mathbf{y}_2}^{-1}(i) \\ \mathbf{R}_{\mathbf{y}_2\mathbf{y}_1}(i)\mathbf{R}_{\mathbf{y}_1\mathbf{y}_1}^{-1}(i) & \mathbf{0} \end{bmatrix}. \tag{10.38}$$

This generic SOS algorithm leads to very robust practical solutions even for a large number of filter taps (see below) due to an inherent normalization by the auto-correlation matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_p}$ as known from the recursive least-squares (RLS) algorithm in supervised adaptive filtering [35]. Again, it is important to note that the products of Sylvester matrices $\mathbf{W}_{pq}$ and the remaining matrices in the update equation (10.38) can be described by linear convolutions. Thus they can be efficiently implemented by a fast convolution as in [25].

Moreover, by comparing (10.38) to the HOS-SIRP update (10.30), it can be seen that due to the fact that only SOS are utilized we obtain the same update with the nonlinearity omitted, i.e., $\phi_{q,D}(u_q) = 1, q = 1,\ldots,P$.

The original derivation [26] of the generic SOS natural gradient update (10.36) was based on a generalization of the cost function of [10]:

$$\mathcal{J}(m) = \sum_{i=0}^{\infty} \beta(i,m)\left\{\log\det\operatorname{bdiag}\mathbf{R}_{\mathbf{yy}}(i) - \log\det\mathbf{R}_{\mathbf{yy}}(i)\right\}. \tag{10.39}$$

In Fig. 10.2 the mechanism of the SOS cost function (10.39) is illustrated. By minimizing $\mathcal{J}(m)$, all cross-correlations for *D* time-lags are reduced and will ideally vanish, while the auto-correlations are untouched. As both cost functions (10.17) and (10.39) lead to the same result in the SOS case, we may now conclude that the algorithm in [26] is in fact the optimum SOS algorithm for convolutive mixtures in the sense of minimum mutual information or ML, which also implies asymptotic Fisher-efficiency [1, 39].

Another interesting finding is that for both, the holonomic and nonholonomic versions of the HOS update (10.25), (10.26), the SOS BSS algorithm obtained by inserting the Gaussian pdf (10.34) turns out to be nonholonomic confirming its good performance for speech sources.

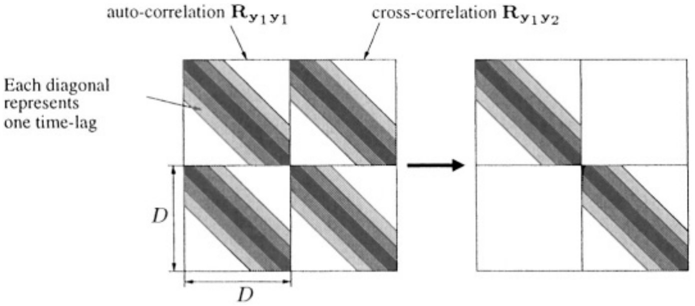

auto-correlation $\mathbf{R}_{\mathbf{y}_1\mathbf{y}_1}$
cross-correlation $\mathbf{R}_{\mathbf{y}_1\mathbf{y}_2}$
Each diagonal represents one time-lag
D
D

Figure 10.2 Illustration of the SOS cost function (10.39).

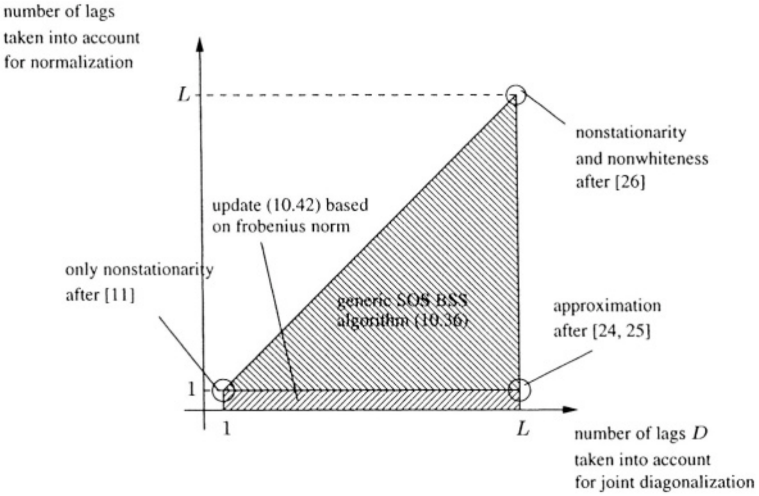

number of lags taken into account for normalization
L
1
1
L
number of lags D taken into account for joint diagonalization
nonstationarity and nonwhiteness after [26]
update (10.42) based on frobenius norm
only nonstationarity after [11]
generic SOS BSS algorithm (10.36)
approximation after [24, 25]

Figure 10.3 Overview of time-domain algorithms based on second-order statistics.

Note that in principle, there are two basic methods to estimate the output correlation matrices (10.37) for nonstationary output signals: the so-called correlation method, and the covariance method as they are known from linear prediction problems [40]. While the correlation method leads to a slightly lower computational complexity due to the Toeplitz structure of the matrices $\mathbf{R}_{\mathbf{yy}}$ (and to smaller matrices, when implemented in the frequency domain covered in Sect. 3.), we consider the more accurate covariance method in this chapter. Note also that (10.37) is full rank since in general we assume $N > PD$.

2.4.3 Approximations of the Generic BSS Based on Second-Order Statistics. The generic update (10.36) is now analyzed and links to known algorithms (see Fig. 10.3) are presented. We highlight here two realizations.

For $D = 1$, the correlation matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_q}(i)$ become scalar values as only a single lag is considered for the correlations. Thus the resulting algorithm is only taking the nonstationarity property into account. This was first proposed by Kawamoto et al. in [11].

In [24, 25], a time-domain algorithm was presented that copes very well with reverberant acoustic environments. Although it was originally introduced as a heuristic extension of [11] incorporating several time-lags, this algorithm can be directly obtained from (10.38) for $D = L$ by approximating the auto-correlation matrices $\mathbf{R}_{\mathbf{y}_q\mathbf{y}_q}(i)$ by the output signal powers, i.e.,

$$\bar{\mathbf{R}}_{\mathbf{y}_q\mathbf{y}_q}(i) = \frac{1}{N}\bar{\mathbf{y}}_q^H(i)\bar{\mathbf{y}}_q(i)\mathbf{I}_{D\times D} \qquad (10.40)$$

for $q = 1, \ldots, P$, where $\bar{\mathbf{y}}_q(\cdot)$ denotes the first column of $\mathbf{Y}_q(\cdot)$. Thus, this approximation is comparable to the well-known normalized least mean squares (NLMS) algorithm in supervised adaptive filtering approximating the RLS algorithm [35]. In addition to the reduced computational complexity, we can ensure the Sylvester structure of the update by using the correlation method [40] for calculation of the short-time correlation matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_q}(i)$ resulting in Toeplitz matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_q}(i)$. The remaining products of Sylvester matrices and Toeplitz matrices in the update equation (10.38) can again be efficiently implemented by a (fast) convolution as was done in [25].

Another very popular subclass of second-order BSS algorithms, particularly for instantaneous mixtures, is based on a cost function using the Frobenius norm $\|\mathbf{A}\|_{\mathrm{F}}^2 = \sum_{i,j} a_{ij}^2$ of a matrix $\mathbf{A} = (a_{ij})$, e.g., [1, 7],[12]-[15]. Analogously to (10.39), this approach may be generalized for convolutive mixtures to

$$\mathcal{J}_{\mathrm{F}}(m) = \sum_{i=0}^{\infty} \beta(i,m) \left\|\mathbf{Y}^H(i)\mathbf{Y}(i) - \operatorname{bdiag}\mathbf{Y}^H(i)\mathbf{Y}(i)\right\|_{\mathrm{F}}^2, \qquad (10.41)$$

which leads (after taking the natural gradient w.r.t. $\mathbf{W}$ in a similar way as in [26]) to the following update equation:

$$\nabla_{\mathbf{W}}^{\mathrm{NG}}\mathcal{J}(m) = 2\sum_{i=0}^{\infty} \beta(i,m)\mathbf{W}\mathbf{R}_{\mathbf{yy}}(i) \begin{bmatrix} \mathbf{0} & \mathbf{R}_{\mathbf{y}_1\mathbf{y}_2}(i) \\ \mathbf{R}_{\mathbf{y}_2\mathbf{y}_1}(i) & \mathbf{0} \end{bmatrix}. \qquad (10.42)$$

We see that this update equation differs from the more general equation (10.38) mainly in the inherent normalization expressed by the inverse matrices $\mathbf{R}_{\mathbf{y}_p\mathbf{y}_p}^{-1}$. Thus, (10.42) can be regarded as an analogon to the least mean square (LMS) algorithm [35] in supervised adaptive filtering. However, many simulation results have shown that for large filter lengths L, (10.42) is prone to instability, while (10.38) shows a very robust convergence behaviour (see Sect. 5.) even for hundreds or thousands of filter coefficients in BSS for real acoustic environments.

3. GENERIC FREQUENCY-DOMAIN BSS ALGORITHM

Frequency-domain BSS is very popular for convolutive BSS since all techniques originally developed for instantaneous BSS can be applied independently in each frequency bin in the discrete Fourier transform (DFT) domain. Furthermore, the fast Fourier transform (FFT) can be used for an efficient implementation. Such narrowband approaches can be found, e.g., in [1], [3], [6], [9], [12]-[17], [22]. Unfortunately, the permutation problem, which is inherent in BSS (e.g., [1]), may then also appear independently in each frequency bin so that extra measures have to be taken to avoid this *internal* permutation. Additionally, as discussed in Sections 2.3 and 2.4 the products involving Sylvester matrices in the time-domain update equations correspond to linear convolutions. Thus, in the narrowband frequency-domain approach these convolutions become circular ones. The resulting wrap-around effects may limit the separation performance. Based on the above matrix formulation in the time domain, the following derivation of broadband frequency-domain algorithms shows explicitly the relation between time-domain and traditional frequency-domain algorithms, as well as some extensions. In contrast to the narrowband approach this inherently resolves the permutation ambiguity and prevents circular convolution effects in the update equation. Moreover, as in the time-domain, (10.17) also leads to the very desirable property of an inherent stepsize normalization in the frequency domain which also becomes clear by a link with [17] for the SOS case. As pointed out in the previous section, the conditions for the parameters *L, N,* and *D* for the natural gradient adaptation are given by the relations $N > D$ and $1 \leq D \leq L$. Therefore, we may assume $N = L$ without loss of generality for the following derivation.

3.1 GENERAL FREQUENCY-DOMAIN FORMULATION

The matrix formulation (10.11) introduced for the time-domain in Sect. 2. allows a rigorous derivation of the corresponding frequency-domain BSS algorithms. In the frequency domain, the structure of the algorithm depends on the method chosen for estimating the correlation matrices. Here, we consider again the more accurate covariance method [40] (see Sect. 2.4.2). The matrices $\mathbf{X}_p(m)$ and $\mathbf{W}_{pq}$, introduced in Sect. 2.1 are now diagonalized in two steps to Obtain frequency-domain representations. In the following, we mark frequency-domain quantities by an underbar. This does, however, not imply that they are simply DFTs of the corresponding time-domain quantities. Each quantity has to be transformed individually. We first consider the $L \times 2L$ Toeplitz matrices $\mathbf{X}_p(m)$.

Step 1: Transformation of Toeplitz matrices into circulant matrices. Any Toeplitz matrix $\mathbf{X}_p$ (10.16) can be transformed, by doubling its size, to a circulant matrix $\mathbf{C}_{X_p}(m)$ [31]. In our case we define the $4L \times 4L$ circulant matrix by taking into account (10.16) by

$$\mathbf{C}_{X_p}(m) = \begin{bmatrix} \mathbf{X}'_p(m-3) & \mathbf{X}_p(m-1) \\ \mathbf{X}_p(m-2) & \mathbf{X}_p(m) \\ \mathbf{X}_p(m-1) & \mathbf{X}'_p(m-3) \\ \mathbf{X}_p(m) & \mathbf{X}_p(m-2) \end{bmatrix},$$

where $\mathbf{X}'_p(m-3)$ is a properly chosen extension ensuring a circular shift of the $4L$ input values in the first column. It follows

$$\mathbf{X}_p(m) = \mathbf{W}^{01_L}_{L\times 4L}\mathbf{C}_{X_p}(m)\mathbf{W}^{1_{2L}0}_{4L\times 2L}, \tag{10.43}$$

where we introduced the windowing matrices

$$\begin{aligned} \mathbf{W}^{01_L}_{L\times 4L} &= [\mathbf{0}_{L\times 3L}, \mathbf{I}_{L\times L}], \\ \mathbf{W}^{1_{2L}0}_{4L\times 2L} &= [\mathbf{I}_{2L\times 2L}, \mathbf{0}_{2L\times 2L}]^T. \end{aligned}$$

This notation follows the conventions listed in Sect. 2.2.

Step 2: Transformation of the circulant matrices into diagonal matrices. Using the $4L \times 4L$ DFT matrix $\mathbf{F}_{4L\times 4L}$, the circulant matrices are diagonalized as follows:

$$\mathbf{C}_{X_p}(m) = \mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{X}}_p(m)\mathbf{F}_{4L\times 4L},$$

where the diagonal matrices $\underline{\mathbf{X}}_p(m)$ representing the frequency-domain versions of $\mathbf{X}_p(m)$, can be expressed by the first columns of $\mathbf{C}_{X_p}(m)$,

$$\begin{aligned} \underline{\mathbf{X}}_p(m) = {} & \mathrm{diag}\{\mathbf{F}_{4L\times 4L}[x_p(mL-3L), \ldots, x_p(mL-1), \\ & x_p(mL), x_p(mL+1), \ldots, x_p(mL+L-1)]^T\}, \end{aligned} \tag{10.44}$$

i.e., to obtain $\underline{\mathbf{X}}_p(m)$, we transform the concatenated vectors of the current block and three previous blocks of the input signals $x_p(n)$. Here, $\mathrm{diag}\{\mathbf{a}\}$ denotes a square matrix with the elements of vector a on its main diagonal. Now, (10.43) can be rewritten equivalently as

$$\mathbf{X}_p(m) = \mathbf{W}^{01_L}_{L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{X}}_p(m)\mathbf{F}_{4L\times 4L}\mathbf{W}^{1_{2L}0}_{4L\times 2L}. \tag{10.45}$$

Equations (10.44) and (10.45) exhibit a form that is structurally similar to that of the corresponding counterparts of the well-known (supervised) frequency-domain adaptive filters [31]. However, the major difference here is that we need a transformation length of at least $4L$ instead of $2L$ for an accurate broadband formulation. This should come as no surprise, since in BSS using the covariance

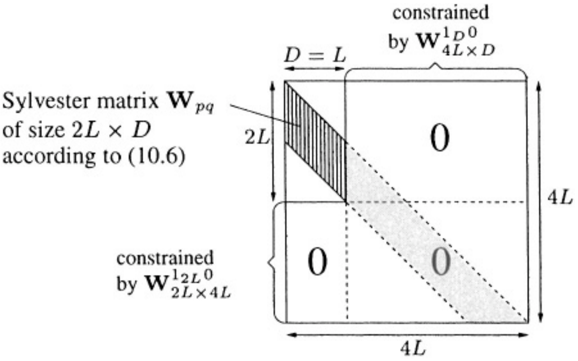

constrained
by $\mathbf{W}^{1_D 0}_{4L\times D}$
$D = L$
Sylvester matrix $\mathbf{W}_{pq}$
of size $2L \times D$
according to (10.6)
$2L$
0
$4L$
constrained
by $\mathbf{W}^{1_{2L} 0}_{2L\times 4L}$
0
0
$4L$

Figure 10.4 Illustration of equation (10.46).

method, both convolution and correlation is carried out where both operations double the transformation length.

We now transform the matrices $\mathbf{W}_{pq}$ in the same way as shown above for $\mathbf{X}_p$. Thereby, we obtain

$$\mathbf{W}_{pq} = \mathbf{W}^{1_{2L}0}_{2L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{W}}_{pq}\mathbf{F}_{4L\times 4L}\mathbf{W}^{1_D 0}_{4L\times D}, \tag{10.46}$$

where

$$\begin{aligned}\mathbf{W}^{1_D 0}_{4L\times D} &= [\mathbf{I}_{D\times D}, \mathbf{0}_{D\times(4L-D)}]^T, \\ \mathbf{W}^{1_{2L}0}_{2L\times 4L} &= [\mathbf{I}_{2L\times 2L}, \mathbf{0}_{2L\times 2L}] = \left(\mathbf{W}^{1_{2L}0}_{4L\times 2L}\right)^T,\end{aligned}$$

and the frequency-domain representation of the demixing matrix

$$\underline{\mathbf{W}}_{pq} = \text{diag}\{\mathbf{F}_{4L\times 4L}[w_{pq,0}, \ldots, w_{pq,L-1}, 0, \ldots, 0]^T\}. \tag{10.47}$$

Equation (10.46) is illustrated in Fig. 10.4. Note that the column vector in (10.47) corresponds to the first column of the $4L \times 4L$ matrix $\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{W}}_{pq}\mathbf{F}_{4L\times 4L}$ in Fig. 10.4. Moreover, it can be seen that the premultiplied transformation $\mathbf{W}^{1_{2L}0}_{2L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}$ in (10.46) is related to the demixing filter taps in the first column of $\mathbf{W}_{pq}$, while the post-multiplied transformation in (10.46), which we denote by

$$\mathbf{L}^{1_D 0}_{4L\times D} = \mathbf{F}_{4L\times 4L}\mathbf{W}^{1_D 0}_{4L\times D}, \tag{10.48}$$

is related to the introduction of D time-lags (see also Sect. 3.3.1). Combining all channels, we obtain from (10.45) and (10.46)

$$\begin{aligned}\mathbf{X}(m) &= \mathbf{W}^{01_L}_{L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{X}}(m) \\ &\quad \cdot \text{bdiag}\{\mathbf{F}_{4L\times 4L}\mathbf{W}^{1_{2L}0}_{4L\times 2L}, \ldots, \mathbf{F}_{4L\times 4L}\mathbf{W}^{1_{2L}0}_{4L\times 2L}\}, \quad (10.49) \\ \mathbf{W} &= \text{bdiag}\{\mathbf{W}^{1_{2L}0}_{2L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}, \ldots, \mathbf{W}^{1_{2L}0}_{2L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\}\underline{\mathbf{W}}\mathbf{L},\end{aligned}$$
$$(10.50)$$

where $\underline{\mathbf{X}}(m)$ and $\underline{\mathrm{W}}$ are defined analogously to (10.13) and (10.10), respectively. $\mathbf{L}$ denotes the $4LP \times DP$ matrix

$$\mathbf{L} = \mathrm{bdiag}\{\mathbf{L}^{1_D0}_{4L\times D}, \dots, \mathbf{L}^{1_D0}_{4L\times D}\}.$$

From (10.11), (10.49), and (10.50) we further obtain

$$\mathbf{Y}(m) = \mathbf{W}^{01_L}_{L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{Y}}(m)\mathbf{L}, \tag{10.51}$$

with

$$\underline{\mathbf{Y}}(m) = \underline{\mathbf{X}}(m)\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{W}}, \tag{10.52}$$

and the time-domain constraints

$$\begin{aligned}
\mathbf{G}^{1_{2L}0}_{4LP\times 4LP} &= \mathrm{bdiag}\left\{\mathbf{G}^{1_{2L}0}_{4L\times 4L}, \dots, \mathbf{G}^{1_{2L}0}_{4L\times 4L}\right\},\\
\mathbf{G}^{1_{2L}0}_{4L\times 4L} &= \mathbf{F}_{4L\times 4L}\mathbf{W}^{1_{2L}0}_{4L\times 4L}\mathbf{F}^{-1}_{4L\times 4L},\\
\mathbf{W}^{1_{2L}0}_{4L\times 4L} &= \mathbf{W}^{1_{2L}0}_{4L\times 2L}\mathbf{W}^{1_{2L}0}_{2L\times 4L}\\
&= \begin{bmatrix} \mathbf{I}_{2L\times 2L} & \mathbf{0}_{2L\times 2L}\\ \mathbf{0}_{2L\times 2L} & \mathbf{0}_{2L\times 2L}\end{bmatrix}.
\end{aligned}$$

To formulate the cost function (10.17) in the frequency domain, we first need to express it *equivalently* using matrices $\mathbf{Y}_p$, $p = 1, \dots, P$. This inevitably leads to the introduction of pdfs which depend on matrices in their arguments. In general, such pdfs are determined by a fourth-order tensor which contains all cross-relations between the matrix elements. However, due to the Toeplitz structure of the matrices $\mathbf{Y}_p$ a redundancy is introduced which neither appears in the cost function (10.17) nor leads to any improved results compared to (10.17).

Thus, we can replace the tensor by a matrix containing only the desired information on the cross-relations between the D time-lags. This yields the following equivalent representation of (10.17):

$$\begin{aligned}
\mathcal{J}(m) &= -\sum_{i=0}^{\infty}\beta(i,m)\frac{1}{N}\left\{\log\left(\tilde{p}_{1,N\times D}(\mathbf{Y}_1(i))\cdot\ldots\cdot\tilde{p}_{P,N\times D}(\mathbf{Y}_P(i))\right)\right.\\
&\quad \left.-\log\left(\tilde{p}_{N\times PD}(\mathbf{Y}_1(i),\dots,\mathbf{Y}_P(i))\right)\right\},
\end{aligned} \tag{10.53}$$

with the *auxiliary pdfs* which we define here by

$$\tilde{p}_{p,N\times D}(\mathbf{Y}_p(i)) = \prod_{j=0}^{N-1}\hat{p}_{p,D}(\mathbf{y}_p(i,j)), \tag{10.54}$$

$$\tilde{p}_{N\times PD}(\mathbf{Y}_1(i),\dots,\mathbf{Y}_P(i)) = \prod_{j=0}^{N-1}\hat{p}_{PD}(\mathbf{y}_1(i,j),\dots,\mathbf{y}_P(i,j)), \tag{10.55}$$

showing the relation to the multivariate pdfs. The equivalence to (10.17) can easily be verified by inserting (10.54) and (10.55) in (10.53). The advantage of introducing such auxiliary pdfs is that they can formally be handled like standard pdfs where the rows of the matrix in their argument are mutually statistically independent. This allows a compact representation of the following equations.

To proceed with the derivation, we take the gradient of (10.53) w.r.t. the frequency-domain coefficient matrix $\underline{\mathbf{W}}$. This is done analogously to the time-domain derivation of (10.19). However, (10.51) and (10.52) have to be taken into account by using the chain rule for matrices [41]. This finally leads to the following gradient for the frequency-domain update:

$$\begin{aligned}\nabla_{\underline{\mathbf{W}}}\mathcal{J}(m) &= \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\left\{\mathbf{G}_{4LP\times 4LP}^{1_{2L}0}\underline{\mathbf{X}}^H(i)\underline{\mathbf{\Phi}}(\underline{\mathbf{Y}}(i))\right.\\ &\quad \left.-\mathbf{L}\left(\mathbf{L}^H\underline{\mathbf{W}}^H\mathbf{L}\right)^{-1}\mathbf{L}^H\right\} \qquad (10.56)\end{aligned}$$

with the frequency-domain score function

$$\underline{\mathbf{\Phi}}(\underline{\mathbf{Y}}(i)) = \left[-\frac{\frac{\partial \tilde{\underline{p}}_{1,4L\times 4L}(\underline{\mathbf{Y}}_1(i))}{\partial \underline{\mathbf{Y}}_1(i)}}{\tilde{\underline{p}}_{1,4L\times 4L}(\underline{\mathbf{Y}}_1(i))}, \dots, -\frac{\frac{\partial \tilde{\underline{p}}_{P,4L\times 4L}(\underline{\mathbf{Y}}_P(i))}{\partial \underline{\mathbf{Y}}_P(i)}}{\tilde{\underline{p}}_{P,4L\times 4L}(\underline{\mathbf{Y}}_P(i))}\right]. \qquad (10.57)$$

Note that the pdf $\tilde{\underline{p}}_{q,4L\times 4L}(\underline{\mathbf{Y}}_q(i))$ of the frequency-domain matrix $\underline{\mathbf{Y}}_q(i)$ is obtained by transforming the pdf $\tilde{p}_{q,N\times D}(\mathbf{Y}_q(i))$ of time-domain variables using (10.51). We will go into the precise formulation of $\tilde{\underline{p}}_{q,4L\times 4L}(\underline{\mathbf{Y}}_q(i))$ within the scope of the special cases treated in Sect. 3.3. Equations (10.56) and (10.57) are the generic frequency-domain counterparts of (10.19) and (10.20), respectively, and may be equivalently used for coefficient adaptation.

As in the time-domain, we need not calculate the entire coefficient matrix $\underline{\mathbf{W}}$ explicitly due to the redundancy introduced by the Sylvester structure in the time domain, and the diagonal structure of the submatrices $\underline{\mathbf{W}}_{pq}$ in the frequency domain, respectively. While the structure of matrix $\underline{\mathbf{W}}$ is independent of *D*, matrix $\mathbf{L}$ introduces the number of time-lags taken into account by the cost function, as shown by (10.48) and (10.51) (see also Fig. 10.4). To calculate the separated output signals, given a demixing matrix $\underline{\mathbf{W}}$, we need to pick the first column of $\mathbf{Y}$ in (10.51) (the other columns were introduced in (10.11) for including multiple time-lags in the cost function). This is done by using $\mathbf{L} = \mathbf{L}_{\mathrm{I}} = \mathrm{bdiag}\{\mathbf{1}_{4L\times 1}, \cdots, \mathbf{1}_{4L\times 1}\}$ in (10.51). Then, $\underline{\mathbf{W}}\mathbf{L}$ in that equation becomes a $4LP \times P$ matrix $\underline{\mathbf{W}}'$ whose columns correspond to the diagonals of $\underline{\mathbf{W}}$. As a general rule,

$$\underline{\mathbf{W}}' = \underline{\mathbf{W}}\mathbf{L}_{\mathrm{I}}, \qquad (10.58)$$

and building diagonal submatrices $\underline{\mathbf{W}}_{pq}$ of $\underline{\mathbf{W}}$ using the entries of $\underline{\mathbf{W}}'$, transforms the two equivalent representations into each other. Thus, to formally obtain the update of $\underline{\mathbf{W}}'$ needed for the output signal calculation, we post-multiply (10.56) by $\mathbf{L}_{\mathrm{I}}$, both simplifying the calculation of (10.56), and enforcing the diagonal structure of $\underline{\mathbf{W}}_{pq}$ during the adaptation. This simplification results from the fact that we only have to operate with vectors rather than matrices for each channel when constructing the update equation from the right to the left in a practical realization.

In addition to the diagonal structure of $\underline{\mathbf{W}}$, we have to ensure the Sylvester structure in the time domain as noted previously. As can be seen in Fig. 10.4, (10.47) determines the first column, and thus the whole $4L \times 4L$ Sylvester matrix. In other words, we have to ensure that the time-domain column vector in (10.47) contains only L filter coefficients and $3L$ zeros. Therefore, the gradient (10.56) has to be constrained by $\mathbf{G}^{1L0}_{4LP\times 4LP}$. Together with (10.58) this leads to

$$\Delta\underline{\mathbf{W}}'(m) = \mathbf{G}^{1L0}_{4LP\times 4LP}\nabla_{\underline{\mathbf{W}}}\mathcal{J}(m)\mathbf{L}_{\mathrm{I}}, \tag{10.59}$$

which may again be implemented efficiently from the right to the left. Then the constraint $\mathbf{G}^{1L0}_{4LP\times 4LP}$ reduces to channel-wise inverse FFT, windowing (see also Sect. 3.3.2), and FFT operations.

3.2 NATURAL GRADIENT IN THE FREQUENCY DOMAIN

In Sect. 2.3, it has been shown that the natural gradient for convolutive mixtures introduced there for the time domain yields equivariant adaptation algorithms, i.e., the evolutionary behaviour of

$$\mathbf{C}(m) = \mathbf{H}\mathbf{W}(m) \tag{10.60}$$

and $\Delta\mathbf{C}(m) = \mathbf{H}\Delta\mathbf{W}(m)$ does not explicitly depend on $\mathbf{H}$ in (10.26).

In this section, we investigate how this formulation of the natural gradient transforms into the frequency domain. To begin with, we start by the following approach containing arbitrary matrices $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$, and $\mathbf{A}_4$ of proper size:

$$\nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}}\mathcal{J} = \mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2\mathbf{A}_3\underline{\mathbf{W}}^H\mathbf{A}_4\nabla_{\underline{\mathbf{W}}}\mathcal{J}. \tag{10.61}$$

Now, our task is to determine the four matrices $\mathbf{A}_i$ such that the resulting coefficient update exhibits desired properties.

As a first condition, matrix $\mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2\mathbf{A}_3\underline{\mathbf{W}}^H\mathbf{A}_4$ in (10.61) must be positive definite, i.e., all its eigenvalues must be positive to ensure convergence [39]. This determines matrices $\mathbf{A}_3$, and $\mathbf{A}_4$ up to a positive scalar constant, which can be absorbed in the stepsize, so that we obtain

$$\nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}}\mathcal{J} = \mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2\mathbf{A}_2^H\underline{\mathbf{W}}^H\mathbf{A}_1^H\nabla_{\underline{\mathbf{W}}}\mathcal{J}. \tag{10.62}$$

As the second, and most important condition, it is required that the equivariance property is fulfilled. Combining (10.60) with (10.50), we obtain a relation between $\mathbf{C}$ and the frequency-domain coefficients $\underline{\mathbf{W}}$,

$$\mathbf{C} = \mathbf{H}\,\mathrm{bdiag}\{\cdots\}\underline{\mathbf{W}}\mathbf{L}, \tag{10.63}$$

and analogously

$$\begin{aligned} \Delta\mathbf{C} &= \mathbf{H}\,\mathrm{bdiag}\{\cdots\}\Delta\underline{\mathbf{W}}\mathbf{L} \\ &= \mathbf{H}\,\mathrm{bdiag}\{\cdots\}\nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}}\mathcal{J}\mathbf{L}. \end{aligned} \tag{10.64}$$

As in the time domain (see (10.26)), it is required that (10.64) in combination with the natural gradient (10.62) can be expressed by $\mathbf{C}$ defined in (10.63), and therefore does not explicitly depend on $\mathbf{H}$. This leads to the claim

$$\Delta\mathbf{C} = \underbrace{\mathbf{H}\,\mathrm{bdiag}\{\cdots\}\mathbf{A}_1\underline{\mathbf{W}}\mathbf{A}_2}_{=\mathbf{C}}\,\mathbf{A}_2^H\underline{\mathbf{W}}^H\mathbf{A}_1^H\nabla_{\underline{\mathbf{W}}}\mathcal{J}\mathbf{L}, \tag{10.65}$$

and a comparison of (10.65) with (10.63) yields the matrices

$$\mathbf{A}_1 = \mathbf{G}^{1_{2L}0}_{4LP\times 4LP}, \quad \mathbf{A}_2 = \mathbf{L}. \tag{10.66}$$

Note that $\mathbf{A}_1 = \mathbf{I}$ is not the general solution. This can be verified by inserting (10.66) in (10.65), and considering the argument of bdiag$\{\cdot\}$ acording to (10.50).

Finally, we obtain the natural gradient

$$\begin{aligned} \nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}}\mathcal{J} &= \mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{W}}\mathbf{L}\mathbf{L}^H\underline{\mathbf{W}}^H\left(\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\right)^H\nabla_{\underline{\mathbf{W}}}\mathcal{J} \\ &= \mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{W}}\mathbf{L}\mathbf{L}^H\underline{\mathbf{W}}^H\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\nabla_{\underline{\mathbf{W}}}\mathcal{J}, \end{aligned} \tag{10.67}$$

and together with (10.56) it follows the coefficient update

$$\nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}}\mathcal{J}(m) = \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{W}}\mathbf{L}\mathbf{L}^H\left\{\underline{\mathbf{Y}}^H(i)\underline{\boldsymbol{\Phi}}(\underline{\mathbf{Y}}(i)) - \mathbf{I}\right\}. \tag{10.68}$$

Note that in this equation the natural gradient shows again the convenient property of avoiding one matrix inversion. Formally, as in Sect. 3.1, (10.59) can be used to obtain $\Delta\underline{\mathbf{W}}'$.

3.3 SPECIAL CASES AND LINKS TO KNOWN FREQUENCY-DOMAIN ALGORITHMS

The generic gradient (10.56) and generic natural gradient (10.68), respectively, exhibit three types of quantities that fully specify practical realizations which follow as special cases. These quantities can be related to the three fundamental signal properties, as shown in Table 10.1.

Table 10.1 Quantities defining a certain frequency-domain algorithm.

quantity	related to	examples in
constraints $\mathbf{G}^{\cdots}_{\cdots}$, $\mathbf{L}$	nonwhiteness	Sect. 3.3.1, 3.3.2
score function $\underline{\mathbf{\Phi}}(\cdot)$	nongaussianity	Sect. 3.3.1, 3.3.3
weighting function $\beta(\cdot)$	nonstationarity	Sect. 4.

3.3.1 The Constraints and the Internal Permutation Problem in Frequency-Domain BSS. Two types of constraints appear in the gradient (10.56) and in the natural gradient update (10.68):

- The matrices $\mathbf{G}^{\cdots}_{\cdots}$ in (10.52) and in the update equations are mainly responsible for preventing decoupling of the individual frequency components, and thus avoiding the internal permutation among the different frequency bins and circular convolution effects.

- Matrix $\mathbf{L}$ has two different functions: on the one hand, it allows joint diagonalization over D time-lags, and on the other hand, it acts as time-domain constraint similar to the matrices $\mathbf{G}^{\cdots}_{\cdots}$ (see Fig. 10.4).

Note that the constraints $\mathbf{G}^{\cdots}_{\cdots}$ and $\mathbf{L}$ also appear in the score function $\underline{\mathbf{\Phi}}(\cdot)$ as can be seen later (e.g., Sect. 3.3.3) in more detail.

Concerning matrix $\mathbf{L}$ we can distinguish between four different cases:

a) $D < L$: As in the time domain, this choice allows the exploitation of the nonwhiteness property with up to D time-lags.

b) $D = L$: This is the optimum case as in the time domain.

c) $D > L$: This choice is not meaningful in the time domain. In the frequency domain, however, we can choose D up to the transformation length $4L$ due to the introduced circulant matrix, as shown in Fig. 10.4. For $D > L$ the time-domain constraint is relaxed, which may also lead to a suboptimum solution.

d) $D = 4L$: According to Fig. 10.4 this corresponds to the traditional narrowband approximation (apart from constraints $\mathbf{G}^{\cdots}_{\cdots}$) so that all matrices $\mathbf{L}$ cancel out in the update equations, which can also be verified using (10.48).

Case d), i.e., neglecting matrix L in (10.56) yields a simplified gradient

$$\nabla_{\underline{\mathbf{W}}}\mathcal{J}(m) = \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\left\{\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{X}}^{H}(i)\underline{\mathbf{\Phi}}(\underline{\mathbf{Y}}(i)) - \underline{\mathbf{W}}^{-H}\right\}, \tag{10.69}$$

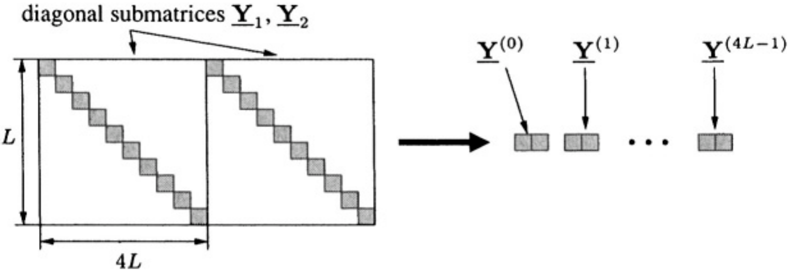

diagonal submatrices $\underline{\mathbf{Y}}_1, \underline{\mathbf{Y}}_2$
4L
4L
$\underline{\mathbf{Y}}^{(0)}$
$\underline{\mathbf{Y}}^{(1)}$
$\underline{\mathbf{Y}}^{(4L-1)}$
• • •

Figure 10.5 Illustration of bin-wise decomposition for the 2-channel case.

where $^{-H}$ denotes the inverse of a conjugate transpose of a matrix, and from (10.68), we obtain a simplified natural gradient

$$\nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}} \mathcal{J}(m) = \frac{2}{N} \sum_{i=0}^{\infty} \beta(i,m) \mathbf{G}^{1_{2L}0}_{4LP \times 4LP} \underline{\mathbf{W}} \left\{ \underline{\mathbf{Y}}^H(i) \underline{\mathbf{\Phi}}(\underline{\mathbf{Y}}(i)) - \mathbf{I} \right\}. \tag{10.70}$$

Note that these expressions still largely avoid the well-known internal permutation problem of frequency-domain BSS using the constraints $\mathbf{G}^{\because}_{\cdots}$ in the calculation of $\underline{\mathbf{Y}}$ in (10.52) and in the update equations obtained from inserting (10.69) or (10.70) in (10.59).

By additionally approximating $\mathbf{G}^{\because}_{\cdots}$ as scaled identity matrices [31] in the gradients, the submatrices $\underline{\mathbf{Y}}_q$ of $\mathbf{Y}$ in (10.69) and (10.70) also become diagonal, as illustrated in Fig. 10.5. Moreover, the frequency-domain multivariate score function $\underline{\mathbf{\Phi}}(\cdot)$ can be decomposed to frequency bin selective score functions $\underline{\mathbf{\Phi}}^{(\nu)}(\cdot)$ containing only univariate pdfs $\tilde{\underline{p}}^{(\nu)}_{p,1}(\cdot)$ for channel p, i.e.,

$$\underline{\mathbf{\Phi}}^{(\nu)}(\underline{\mathbf{Y}}^{(\nu)}(i)) = \left[-\frac{\frac{\partial \tilde{\underline{p}}^{(\nu)}_{1,1}(\underline{Y}^{(\nu)}_1(i))}{\partial \underline{Y}^{(\nu)}_1(i)}}{\tilde{\underline{p}}^{(\nu)}_{1,1}(\underline{Y}^{(\nu)}_1(i))}, \ldots, -\frac{\frac{\partial \tilde{\underline{p}}^{(\nu)}_{P,1}(\underline{Y}^{(\nu)}_P(i))}{\partial \underline{Y}^{(\nu)}_P(i)}}{\tilde{\underline{p}}^{(\nu)}_{P,1}(\underline{Y}^{(\nu)}_P(i))} \right], \tag{10.71}$$

where $\nu = 0, \ldots, 4L-1$ denotes the frequency bin index. This approximation combined with $D = 4L$ (case d) from above) corresponds to the traditional narrowband approach. Only in this case both equations can be decomposed into its frequency components, i.e., we can equivalently write

$$\nabla_{\underline{\mathbf{W}}} \mathcal{J}^{(\nu)}(m) = \frac{2}{N} \sum_{i=0}^{\infty} \beta(i,m) \left\{ \left(\underline{\mathbf{X}}^{(\nu)}(i) \right)^H \underline{\mathbf{\Phi}}^{(\nu)}(\underline{\mathbf{Y}}^{(\nu)}(i)) - \left(\underline{\mathbf{W}}^{(\nu)} \right)^{-H} \right\}, \tag{10.72}$$

and

$$\nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}} \mathcal{J}^{(\nu)}(m) = \frac{2}{N} \sum_{i=0}^{\infty} \beta(i,m) \underline{\mathbf{W}}^{(\nu)} \left\{ \left(\underline{\mathbf{Y}}^{(\nu)}(i) \right)^H \underline{\mathbf{\Phi}}^{(\nu)} (\underline{\mathbf{Y}}^{(\nu)}(i)) - \mathbf{I} \right\}, \tag{10.73}$$

respectively. In contrast to $\underline{\mathbf{W}}$ and $\underline{\mathbf{Y}}$ in (10.56), (10.68) which are $4LP \times 4LP$ and $4L \times 4LP$ matrices, respectively, the corresponding matrices $\underline{\mathbf{W}}^{(\nu)}$ and $\underline{\mathbf{Y}}^{(\nu)}$ in (10.72), (10.73) are only of dimensions $P \times P$ and $1 \times P$, respectively.

The approximation (10.73) of the natural gradient corresponds to the IC A narrowband approach originally proposed by Smaragdis [22] as an extension of the information maximization approach [23].

Note that the nonholonomic version of the natural gradient (10.73) can be obtained similarly to the time domain by replacing matrix $\mathbf{I}$ with $\mathrm{diag}\left\{ \left(\underline{\mathbf{Y}}^{(\nu)}(i) \right)^H \underline{\mathbf{\Phi}}^{(\nu)} \left(\underline{\mathbf{Y}}^{(\nu)}(i) \right) \right\}$.

To derive the update equations from the approximated gradients, we apply again (10.59) which contains another constraint $\mathbf{G}^{1\,L\,0}_{4LP\times 4LP}$ transforming the filter coefficients back into the time domain, zeroing the last $3L$ values, and transforming the result back to the frequency domain. Thus, even if (10.72) and (10.73) can be efficiently computed in a bin-selective manner, this constraint prevents a complete decoupling of the frequency-components in the update equations. This procedure appears similarly in the well-known “constrained frequency-domain adaptive filtering” in the supervised case [35],[31]. In BSS, this theoretically founded mechanism largely eliminates the internal permutation problem in a simple way. It was first heuristically introduced in [22], and also in [14]. A more detailed experimental examination on this constraint was reported in [42] confirming that the ratio between filter length L and transformation length $4L$ - as obtained here analytically - yields optimum separation performance. However, due to the omission of the other constraints in the approximated gradients we will not perfectly remove the permutation ambiguity as observed experimentally in [42]. Traditional narrowband approaches also neglecting the time-domain constraint in (10.59) need additional measures for solving the permutation problem (e.g., [13], [43]).

3.3.2 Alternative Approximations of the Constraints. The generic algorithm (10.68) with its constraint matrices $\mathbf{G}^{:::}_{:::}$ suggests alternative efficient approximations to allow improved tradeoffs between the exact broadband approach (large computational complexity) and the narrowband approach (internal permutation ambiguity) by choosing certain efficient approximations of the constraints.

Generally, we can distinguish between approximations depending on the block index and approximations within each block. One example for the former

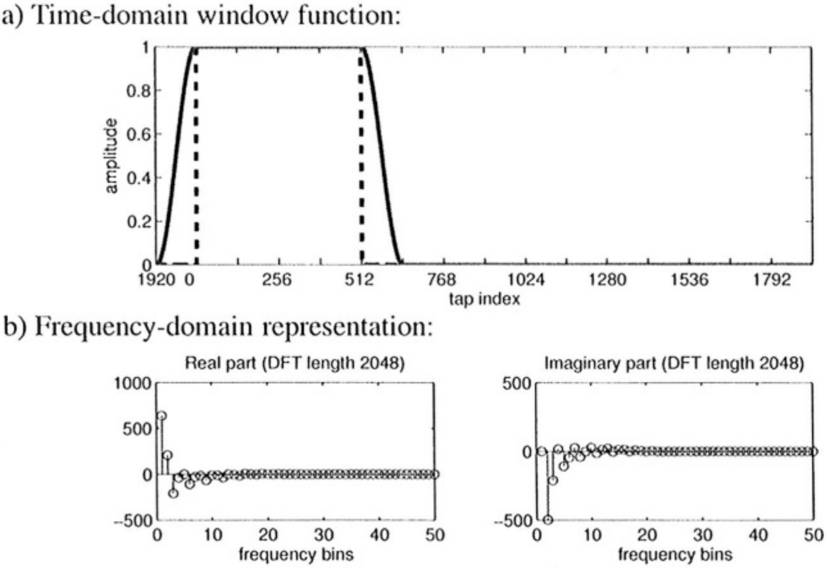


Figure 10.6 Illustration of a smoothed window function for $L = 512$, i.e., transformation length 2048. Note that the window functions are circular.

class is to simply apply the constraints periodically for a reduced number of blocks which has also been proposed for the supervised case [44].

The other class is based on efficient approximations of the rectangular window appearing in the constraints. This is done by smoothing the rectangular window (Fig 10.6a) so that its frequency-domain representation can be well-described by a small number of coefficients (Fig 10.6b). Having such a representation, it is often more efficient to directly apply the convolution operation in the frequency-domain instead of going back and forth between the time domain and frequency domain. This general idea has been discussed earlier for supervised adaptive filtering [45], especially after the introduction of the supervised generic frequency-domain framework [46, 31], see, e.g., [47, 48]. There are several variations possible to design the smoothed window (see also filter design techniques) [49]. However, the smoothed window has to be flat within the length L (e.g., Tukey window [49]). Otherwise compensation terms are necessary [48]. In BSS, a similar windowing has been proposed heuristically in [50].

3.3.3 Generic Frequency-Domain BSS Based on SOS. As shown for the time domain, we derive a generic SOS algorithm by considering Gaussian pdfs. The corresponding Gaussian auxiliary pdf for matrices in the sense described above is obtained using (10.54). It follows

$$\tilde{p}_{p,N\times D}(\mathbf{Y}_p(i)) = \frac{1}{\sqrt{((2\pi)^D \det \mathrm{R}_{\mathbf{y}_p\mathbf{y}_p}(i))^N}} e^{-\frac{1}{2}\mathrm{tr}\left\{\mathbf{Y}_p(i)\mathbf{R}^{-1}_{\mathbf{y}_p\mathbf{y}_p}(i)\mathbf{Y}_p^H(i)\right\}}. \tag{10.74}$$

Transforming this Gaussian pdf into the pdf for the corresponding frequency domain variables $\underline{\mathbf{Y}}_p$ gives again a Gaussian. Using (10.51) and (10.52) we obtain

$$\begin{aligned} \underline{\tilde{p}}_{p,4L\times 4L}(\underline{\mathbf{Y}}_p(i)) \quad \propto \quad & \exp\left\{-\frac{1}{2}\mathrm{tr}\left\{\mathbf{W}^{01_L}_{L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}\underline{\mathbf{Y}}_p(i)\mathbf{L}^{1_D 0}_{L\times D}\mathbf{R}^{-1}_{\mathbf{y}_p\mathbf{y}_p}(i)\right.\right. \\ & \left.\left. \cdot\,(\mathbf{L}^{1_D 0}_{L\times D})^H \underline{\mathbf{Y}}_p^H(i)\mathbf{F}_{4L\times 4L}\mathbf{W}^{01_L}_{4L\times L}\right\}\right\}, \end{aligned} \tag{10.75}$$

where

$$\begin{aligned} \mathbf{R}_{\mathbf{yy}}(i) &= \frac{1}{N}\mathbf{Y}^H(i)\mathbf{Y}(i) = \frac{1}{N}\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}(i)\mathbf{L}, \\ \mathbf{S}_{\mathbf{yy}}(i) &= \underline{\mathbf{Y}}^H(i)\mathbf{G}^{01_L}_{4L\times 4L}\underline{\mathbf{Y}}(i) \\ &= \underline{\mathbf{W}}^H\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\mathbf{S}_{\mathbf{xx}}(i)\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{W}}, \qquad (10.76) \\ \mathbf{S}_{\mathbf{xx}}(i) &= \underline{\mathbf{X}}^H(i)\mathbf{G}^{01_L}_{4L\times 4L}\underline{\mathbf{X}}(i), \qquad (10.77) \\ \mathbf{G}^{01_L}_{4L\times 4L} &= \mathbf{F}_{4L\times 4L}\mathbf{W}^{01_L}_{4L\times 4L}\mathbf{F}^{-1}_{4L\times 4L}, \\ \mathbf{W}^{01_L}_{4L\times 4L} &= \mathbf{W}^{01_L}_{4L\times L}\mathbf{W}^{01_L}_{L\times 4L} \\ &= \begin{bmatrix} \mathbf{0}_{3L\times 3L} & \mathbf{0}_{3L\times L} \\ \mathbf{0}_{L\times 3L} & \mathbf{I}_{L\times L} \end{bmatrix}. \end{aligned}$$

The resulting score function (10.57) reads

$$\underline{\mathbf{\Phi}}(\underline{\mathbf{Y}}(i)) = -\mathbf{G}^{01_L}_{4L\times 4L}\underline{\mathbf{Y}}(i)\mathbf{L}\cdot\mathrm{bdiag}^{-1}\left(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}(i)\mathbf{L}\right)\mathbf{L}^H. \tag{10.78}$$

This leads to

$$\begin{aligned} \nabla_{\underline{\mathbf{W}}}\mathcal{J}(m) &= \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\mathbf{S}_{\mathbf{xy}}\mathbf{L} \\ &\quad \cdot(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L})^{-1}\mathbf{L}^H\left\{\mathbf{S}_{\mathbf{yy}} - \mathrm{bdiag}\,\mathbf{S}_{\mathbf{yy}}\right\}\mathbf{L} \\ &\quad \cdot\mathrm{bdiag}^{-1}\left(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L}\right)\cdot\mathbf{L}^H, \end{aligned} \tag{10.79}$$

where

$$\mathbf{S}_{\mathbf{xy}}(i) = \mathbf{S}_{\mathbf{xx}}(i)\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{W}}. \tag{10.80}$$

Finally with (10.67), we obtain the natural gradient

$$\begin{aligned} \nabla^{\mathrm{NG}}_{\underline{\mathbf{W}}}\mathcal{J}(m) &= \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\mathbf{G}^{1_{2L}0}_{4LP\times 4LP}\underline{\mathbf{W}}\mathbf{L}\mathbf{L}^H\left\{\mathbf{S}_{\mathbf{yy}} - \mathrm{bdiag}\,\mathbf{S}_{\mathbf{yy}}\right\}\mathbf{L} \\ &\quad \cdot\mathrm{bdiag}^{-1}(\mathbf{L}^H\mathbf{S}_{\mathbf{yy}}\mathbf{L})\mathbf{L}^H. \end{aligned} \tag{10.81}$$

Equations (10.79) and (10.81) are the SOS analoga to (10.56) and (10.68). In the same way as shown here for the Gaussian case, we could also analogously define auxiliary pdfs for SIRPs (see Sect. 2.4). Note that in (10.81) the natural gradient shows again the convenient property of avoiding one matrix inversion. Formally, as in Sect. 3.1, (10.59) can be used to obtain $\Delta\underline{\mathbf{W}}'$.

3.3.4 Approximation of the Generic Frequency-Domain BSS Based on SOS. In the SOS case we can apply the same approximation steps as discussed for the HOS case in Sect. 3.3.1. By analogously neglecting matrix **L** in (10.79) and (10.81) we obtain a simplified gradient and a simplified natural gradient, respectively, which still largely avoid the internal permutation problem of frequency-domain BSS.

The narrowband approach is obtained by additionally approximating $\mathbf{G}^{::}_{::}$ as scaled identity matrices [31] yielding gradients which can be decomposed in its frequency components, i.e.,

$$\nabla_{\underline{\mathbf{W}}}\mathcal{J}^{(\nu)}(m) = \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\mathbf{S}_{\mathbf{xy}}^{(\nu)}\left(\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right)^{-1}\left\{\mathbf{S}_{\mathbf{yy}}^{(\nu)} - \operatorname{diag}\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right\}\operatorname{diag}^{-1}\mathbf{S}_{\mathbf{yy}}^{(\nu)} \tag{10.82}$$

and

$$\nabla_{\underline{\mathbf{W}}}^{\mathrm{NG}}\mathcal{J}^{(\nu)}(m) = \frac{2}{N}\sum_{i=0}^{\infty}\beta(i,m)\underline{\mathbf{W}}^{(\nu)}\left\{\mathbf{S}_{\mathbf{yy}}^{(\nu)} - \operatorname{diag}\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right\}\operatorname{diag}^{-1}\mathbf{S}_{\mathbf{yy}}^{(\nu)}, \tag{10.83}$$

respectively, where $\nu = 0, \ldots, 4L-1$ denotes the frequency bins. In contrast to $\mathbf{S}_{\mathbf{xy}}$, $\mathbf{S}_{\mathbf{yy}}$,and $\underline{\mathbf{W}}$in (10.79), (10.81) which are $4LP \times 4LP$ matrices each, the corresponding matrices $\mathbf{S}_{\mathbf{xy}}^{(\nu)}$, $\mathbf{S}_{\mathbf{yy}}^{(\nu)}$, and $\underline{\mathbf{W}}^{(\nu)}$ in (10.82), (10.83) are only of dimension $P \times P$.

To obtain the update equations from the approximated gradients, we apply again (10.59) preventing the complete decoupling by the constraint $\mathbf{G}^{1_L 0}_{4LP\times 4LP}$.

The approximated coefficient update (10.82) is directly related to some well-known frequency-domain BSS algorithms. In [16], an algorithm that is similar to (10.82) was derived by directly optimizing a cost function similar to the one in [10] in a bin-wise manner. More recently, Fancourt and Parra proposed in [17] to apply the magnitude-squared coherence

$$|\gamma_{y_p y_q}^{(\nu)}(m)|^2 = \frac{|S_{y_p y_q}^{(\nu)}(m)|^2}{S_{y_p y_p}^{(\nu)}(m)S_{y_q y_q}^{(\nu)}(m)}, \tag{10.84}$$

$p, q \in \{1, 2\}$ as a cost function for frequency-domain BSS, where $S_{y_p y_q}^{(\nu)}(m)$ denotes the (p,q)-th element of $\mathbf{S}_{\mathbf{yy}}^{(\nu)}(m)$, i.e., the power spectral density in the

ν-th bin and block m. The coherence (10.84) has the very desirable property that

$$0 \leq |\gamma_{y_p y_q}^{(\nu)}(m)|^2 \leq 1, \tag{10.85}$$

which directly translates into an inherent stepsize normalization of the corresponding update equation [17]. In particular, $|\gamma_{y_1 y_2}^{(\nu)}(m)|^2 = 0$ if $\mathbf{y}_1$ and $\mathbf{y}_2$ are orthogonal, and $|\gamma_{y_1 y_2}^{(\nu)}(m)|^2 = 1$ when $\mathbf{y}_1 = a\mathbf{y}_2$ for any non-zero complex number a.

Comparing the update equation (10.82) with that derived in [17], we see that an additional approximation of $\left(\mathbf{S}_{\mathbf{yy}}^{(\nu)}\right)^{-1}$ as a diagonal matrix was used in [17], which results in

$$\begin{aligned} \nabla_{\underline{\mathbf{W}}} \mathcal{J}^{(\nu)}(m) &= \frac{2}{N} \sum_{i=0}^{\infty} \beta(i,m) \mathbf{S}_{\mathbf{xy}}^{(\nu)} \operatorname{diag}^{-1} \mathbf{S}_{\mathbf{yy}}^{(\nu)} \\ &\quad \cdot \left\{ \mathbf{S}_{\mathbf{yy}}^{(\nu)} - \operatorname{diag} \mathbf{S}_{\mathbf{yy}}^{(\nu)} \right\} \operatorname{diag}^{-1} \mathbf{S}_{\mathbf{yy}}^{(\nu)}. \end{aligned} \tag{10.86}$$

The coherence function (10.84) applied in [17] can be extended to the case $P > 2$ by using the so-called generalized coherence [32]. In [26] a link between the SOS cost function (10.39) and the generalized coherence was established. This relationship allows a geometric interpretation of (10.39) and shows that this cost function leads to an inherent stepsize normalization for the coefficient updates.

4. WEIGHTING FUNCTION

In the generalized cost functions (10.17) and (10.39) a weighting function $\beta(i,m)$ was introduced with the block time indices i, m to allow different realizations of the algorithms. Based on the cost function we previously derived stochastic and natural gradient update equations in the time domain and frequency domain. Due to the similar structure of these equations, we will now consider only the time domain for simplicity. There, we can express the coefficient update as

$$\Delta \mathbf{W}(m) = \sum_{i=0}^{\infty} \beta(i,m) \mathcal{Q}(i), \tag{10.87}$$

where $\mathcal{Q}(i)$ denotes the term originating from the i-th block. In the following we distinguish three different types of weighting functions $\beta(i,m)$ for off-line, on-line, and block-on-line realizations [28]. The weighting functions have a finite support, and are normalized such that $\sum_{i=0}^{\infty} \beta(i,m) = 1$.

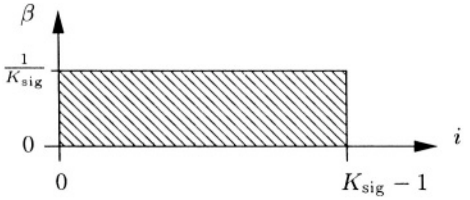


Figure 10.7 Weighting function $\beta(i, m)$ for off-line implementation.

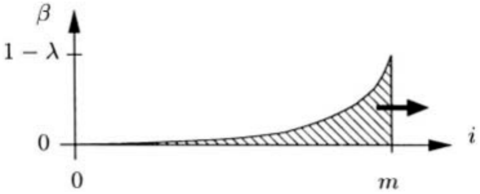


Figure 10.8 Weighting function $\beta(i, m)$ for on-line implementation.

4.1 OFF-LINE IMPLEMENTATION

When realizing the algorithm as an off-line or so-called batch algorithm, then $\beta(i, m)$ corresponds to a rectangular window (Fig. 10.7), which is described by $\beta(i, m) = \frac{1}{K_{\text{sig}}}\epsilon_{0,(K_{\text{sig}}-1)}(i)$, where $\epsilon_{a,b}(i) = 1$ for $a \leq i \leq b$, and $\epsilon_{a,b}(i) = 0$ else. The entire signal is segmented into K_{sig} blocks, and then the entire signal is processed to estimate the demixing matrix $\mathbf{W}^{\ell}$ where the superscript ℓ denotes the current iteration of the coefficient update

$$\mathbf{W}^{\ell} = \mathbf{W}^{\ell-1} - \frac{\mu}{K_{\text{sig}}} \sum_{i=0}^{K_{\text{sig}}-1} \mathcal{Q}(i). \tag{10.88}$$

Hence, the algorithm is generally visiting the signal data repeatedly for each iteration ℓ and therefore it usually achieves a better performance compared to its on-line counterpart.

4.2 ON-LINE IMPLEMENTATION

In time-varying environments an on-line implementation of (10.87) is required. An efficient realization can be achieved by using a weighting function with an exponential forgetting factor λ (Fig. 10.8). It is defined by

$$\beta(i, m) = (1 - \lambda)\lambda^{m-i}\epsilon_{0,m}(i), \tag{10.89}$$

where $0 \leq \lambda < 1$. Thus (10.87) reads

$$\Delta\mathbf{W}(m) = (1 - \lambda)\sum_{i=0}^{m} \lambda^{m-i}\mathcal{Q}(i), \tag{10.90}$$

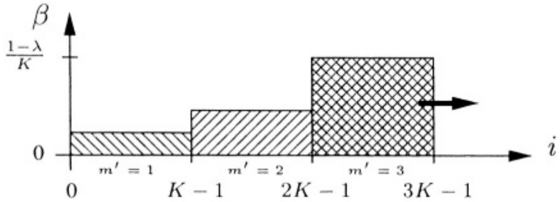


Figure 10.9 Weighting function $\beta(i, m)$ for block-on-linc implementation. Note that $m' = \frac{m}{K}$ denotes the new block index.

where m denotes the current block. Additionally, (10.90) can be formulated recursively to reduce computational complexity and memory requirements since only the preceding demixing matrix has to be saved for the update. This leads to the following coefficient update to be used in (10.23):

$$\Delta\mathbf{W}(m) = \lambda\Delta\mathbf{W}(m-1) + (1-\lambda)\mathcal{Q}(m). \tag{10.91}$$

For the special case $\lambda = 0$ we have

$$\mathbf{W}(m) = \mathbf{W}(m-1) - \mu\mathcal{Q}(m), \tag{10.92}$$

which corresponds to $\beta(i, m) = \delta(i-m)$.

4.3 BLOCK-ON-LINE IMPLEMENTATION

The on-line and off-line approaches can be combined in a so-called block-on-line method (Fig. 10.9) which has been applied for BSS, e.g., in [51]. After obtaining *K* blocks of length *N* we process an off-line algorithm with $\ell_{\max}$ iterations. The demixing filter matrix $\mathbf{W}(m')$ of the current block m' is then used as initial value for the off-line algorithm of the next block. This block-on-line approach allows a tradeoff between computational complexity on the one hand and separation performance and speed of convergence on the other hand by adjusting the maximum number of iterations $\ell_{\max}$ as we will see in Sect. 5..

5. EXPERIMENTS AND RESULTS

Experiments have been conducted using speech data convolved with impulse responses of a real room (580cm × 590cm × 310cm) with a reverberation time $T_{60} = 150\text{ms}$ and a sampling frequency of 16 kHz. A two-element microphone array with an inter-element spacing of 16 cm was used. The speech signals arrived from two different directions, –45° and 45°. The distance between the speakers and the microphones was 2.0m. The length of the source signals (two male speakers from the TIMIT speech corpus [52]) was 10 seconds. The performance was evaluated by means of the signal-to-interference ratio (SIR), defined as the ratio of the signal power of the target signal to the signal power from the jammer signal. For off-line implementations the SIR was calculated

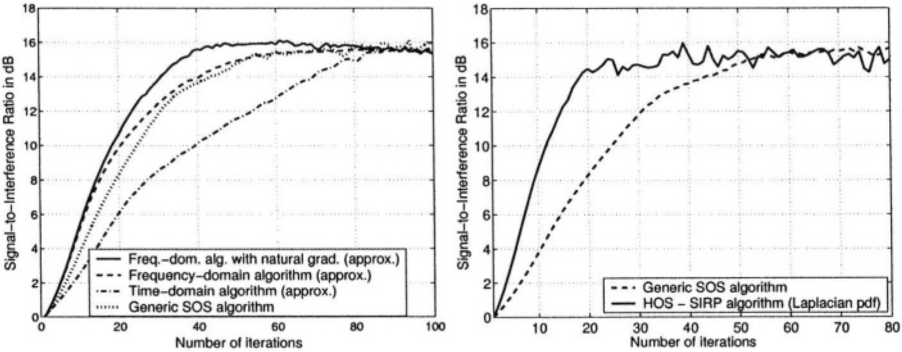


Figure 10.10 Comparison of different off-line realizations (left: SOS algorithms, right: SOS vs. HOS).

over the entire signal length, whereas for on-line implementations it was continuously calculated for each block. In the following the SIR is averaged over both channels.

In our experiments we compared off-line and on-line realizations and we examined the effect of taking into account different numbers of time-lags D for the computation of the correlation function in (10.28) and (10.37). In all experiments the unmixing filter length was set to $L = 512$, the number of lags to $D = 512$, and the block length to $N = 1024$, respectively. Note that the stepsizes of all algorithms have been maximized up to the stability margins.

The framework developed here also allows a better understanding of the initialization of $\mathbf{W}$. It can be shown using (10.6) and (10.25) that the first coefficient of each filter $\mathbf{W}_{pp}$ must be nonzero. This is ensured by using unit impulses for the first filter tap in each$\mathbf{W}_{pp}$. The filters $\mathbf{W}_{pq}$, $p \neq q$ are set to zero.

In the left plot of Fig. 10.10 different off-line SOS algorithms are shown. It can be seen that the approximated gradient (10.82) (dashed) and natural gradient (10.83) (solid) versions of the generic SOS frequency-domain algorithm exhibit the best convergence. This is mainly due to the decomposition of the update equation in its frequency components and hence we have an independent update in each frequency bin. The complete decoupling and therefore also the internal permutation problem is prevented by considering the constraint $\mathbf{G}_{4LP\times 4LP}^{1L0}$ (10.59) (see Sect. 3.1).

It should be pointed out that the generic SOS time-domain algorithm (10.36) (dotted) achieves almost the same convergence as the frequency-domain algorithms. This shows that also time-domain algorithms can exhibit a stable and robust convergence behaviour for long unmixing filters. However, in the generic SOS time-domain algorithm this comes with an increased computational cost, as an inversion of a large matrix is required due to the RLS-like normalization

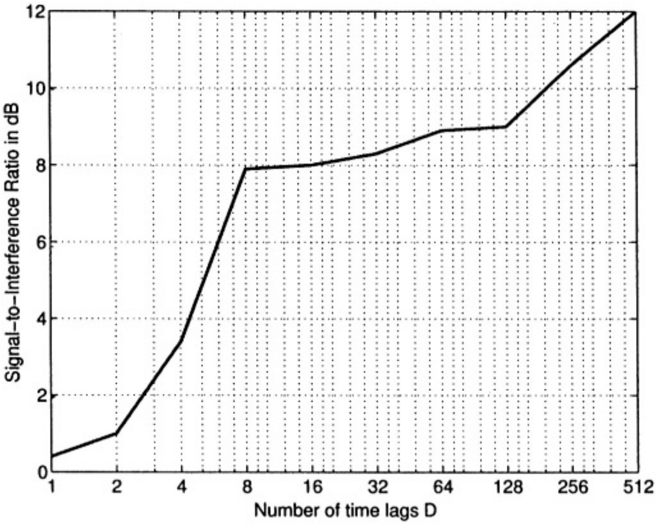


Figure 10.11 Effect of exploiting nonwhiteness by taking into account different numbers of lags D. ($L = 512$)

(see Sect. 2.4.2). The approximated version of the generic SOS time-domain algorithm (dash-dotted) according to (10.40) shows a slower convergence as the RLS-like normalization is replaced by a diagonal matrix which corresponds to an NLMS-like normalization. Moreover it can be seen that all curves converge to the same maximum SIR value which does not depend on the choice of adaptation algorithm.

In the right plot of Fig. 10.10 we compared the generic SOS algorithm in the time-domain and the generic HOS algorithm with the SIRP model from the Laplacian pdf (10.33). Note that the argument of the modified Bessel functions $K_{D/2+1}(\cdot)$ in (10.33) has to be properly regularized. The additional gain in convergence speed of HOS over SOS is due to the additional exploitation of nongaussianity.

In Fig. 10.11 the dependency of the SIR on the number of lags D used for the computation of the correlation function $\mathbf{R}_{\mathbf{yy}}$ in the SOS algorithms is illustrated. An off-line version of the approximated time-domain algorithm (10.40) was evaluated after 50 iterations. We observe a steep increase of the achievable separation performance for up to 8 lags. This can be explained by the fact that speech is strongly correlated within the first lags. By considering these temporal correlations, i.e., nonwhiteness, additional information about the mixtures is taken into account for the simultaneous diagonalization of $\mathbf{R}_{\mathbf{yy}}$. A further increase of D still improves the SIR slightly as the temporal correlation of the room impulse response is considered in the adaptation.

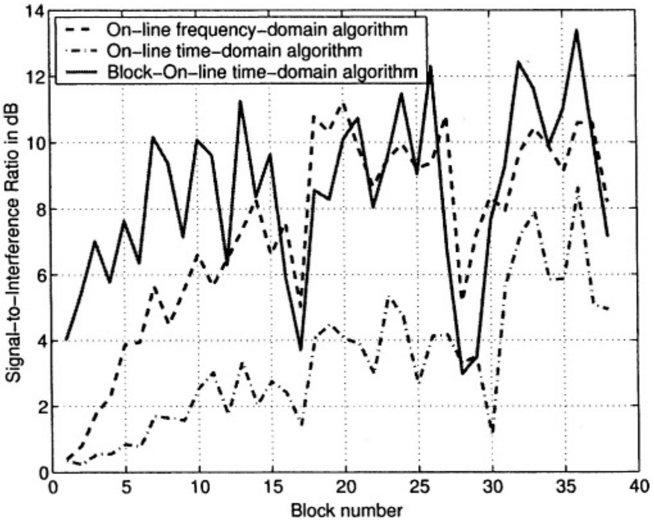


Figure 10.12 Comparison of different on-line realizations.

Various on-line realizations of SOS algorithms are shown in Fig. 10.12. Obviously, the frequency-domain algorithm (dashed) exhibits superior convergence compared to the time-domain algorithm (dash-dotted) due to the NLMS-like approximation of the normalization in the time domain. However, it can also be seen that this effect can be mitigated by using a block-on-line adaptation (see 4.3) (solid) with $K = 8$, $N = 512$, and $\ell_{\max} = 10$ iterations. This leads to improved convergence and separation performance at the expense of increased computational cost.

6. CONCLUSIONS

We presented a unified treatment of BSS algorithms for convolutive mixtures. This framework contains two main principles: Firstly, three fundamental signal properties, nonwhiteness, nonstationarity, and nongaussianity are explicitly taken into account in the generic cost function. Secondly, the framework is based on a general broadband formulation and optimization of this cost function. Due to this approach, rigorous derivations of both known and novel algorithms in the time and frequency domain became possible. Moreover, the introduced matrix formulation with the resulting constraints provides a deeper understanding of the internal permutation ambiguity appearing in traditional narrowband frequency-domain BSS. Experimental results confirm the theoretical findings and demonstrate that this approach allows BSS in both, time and frequency domains for reverberant acoustic environments.

References

[1] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis,* Wiley & Sons, Inc., New York, 2001.

[2] M. Zibulevsky and B.A. Pearlmutter, "Blind source separation by sparse decomposition in a signal dictionary," *Neural Computation,* vol. 13, pp. 863-882, 2001.

[3] S. Araki, S. Makino, A. Blin, R. Mukai, and H. Sawada, "Blind Separation of More Speech than Sensors with Less Distortion by Combining Sparseness and ICA," in *Proc. Int. Workshop on Acoustic Echo and Noise Control (IWAENC),* Kyoto, Japan, Sep. 2003, pp. 271-274.

[4] J.-F. Cardoso and A. Souloumiac, "Blind beamforming for non gaussian signals," *IEE Proceedings-F,* vol 140, no. 6, pp. 362-370, Dec. 1993.

[5] W. Herbordt and W. Kellermann, "Adaptive beamforming for audio signal acquisition," in *Adaptive signal processing: Application to real-world problems,* J. Benesty and Y. Huang, Eds., pp. 155-194, Springer, Berlin, Jan. 2003.

[6] E. Weinstein, M. Feder, and A. Oppenheim, "Multi-channel signal separation by decorrelation," *IEEE Trans. on Speech and Audio Processing,* vol 1, no. 4, pp. 405-413, Oct. 1993.

[7] L. Molgedey and H.G. Schuster, "Separation of a mixture of independent signals using time delayed correlations," *Physical Review Letters,* vol. 72, pp. 3634-3636, 1994.

[8] L. Tong, R.-W. Liu, V.C. Soon, and Y.-F. Huang, "Indeterminacy and identifiability of blind identification," *IEEE Trans. on Circuits and Systems,* vol. 38, pp. 499-509, 1991.

[9] S. Van Gerven and D. Van Compernolle,"Signal separation by symmetric adaptive decorrelation: stability, convergence, and uniqueness," *IEEE Trans. Signal Processing,* vol. 43, no. 7, pp. 1602-1612, 1995.

[10] K. Matsuoka, M. Ohya, and M. Kawamoto, "A neural net for blind separation of nonstationary signals," *Neural Networks,* vol. 8, no. 3, pp. 411-419, 1995.

[11] M. Kawamoto, K. Matsuoka, and N. Ohnishi, "A method of blind separation for convolved non-stationary signals," *Neurocomputing,* vol. 22, pp. 157-171, 1998.

[12] J.-F. Cardoso and A. Souloumiac, "Jacobi angles for simultaneous diagonalization," *SIAM J. Mat. Anal. Appl.,* vol. 17, no. 1, pp. 161-164, Jan. 1996.

[13] S. Ikeda and N. Murata, "An approach to blind source separation of speech signals," *Proc. Int. Symposium on Nonlinear Theory and its Applications,* Crans-Montana, Switzerland, 1998.

[14] L. Parra and C. Spence, "Convolutive blind source separation of non-stationary sources," *IEEE Trans. Speech and Audio Processing,* pp. 320-327, May 2000.

[15] D.W.E. Schobben and P.C.W. Sommen, "A frequency-domain blind signal separation method based on decorrelation," *IEEE Trans on Signal Processing,* vol. 50, no. 8, pp. 1855-1865, Aug. 2002.

[16] H.-C. Wu and J.C. Principe,"Simultaneous diagonalization in the frequency domain (SDIF) for source separation," in *Proc. IEEE Int. Symposium on Independent Component Analysis and Blind Signal Separation (ICA),* 1999, pp. 245-250.

[17] C.L. Fancourt and L. Parra, "The coherence function in blind source separation of convolutive mixtures of non-stationary signals," in Proc. *Int. Workshop on Neural Networks for Signal Processing (NNSP),* 2001.

[18] P. Comon, "Independent component analysis, a new concept? " *Signal Processing,* vol. 36, no. 3, pp. 287-314, Apr. 1994.

[19] A. Cichocki and S. Amari, *Adaptive Blind Signal and Image Processing,* Wiley & Sons, Ltd., Chichester, UK, 2002.

[20] S. Amari, A. Cichocki, and H.H. Yang, "A new learning algorithm for blind signal separation," in *Advances in neural information processing systems,* 8, Cambridge, MA, MIT Press, 1996, pp. 757-763.

[21] J.-F. Cardoso, "Blind signal separation: Statistical principles," *Proc. IEEE,* vol. 86, pp. 2009-2025, Oct. 1998.

[22] P. Smaragdis, "Blind separation of convolved mixtures in the frequency domain," *Neurocomputing,* vol. 22, pp. 21-34, July 1998.

[23] A.J. Bell and T.J. Sejnowski, "An information-maximisation approach to blind separation and blind deconvolution," *Neural Computation,* vol. 7, pp. 1129-1159, 1995.

[24] T. Nishikawa, H. Saruwatari, and K. Shikano, "Comparison of time-domain ICA, frequency-domain ICA and multistage ICA for blind source separation," in *Proc. European Signal Processing Conference (EUSIPCO),* Sep. 2002, vol. 2, pp. 15-18.

[25] R. Aichner, S. Araki, S. Makino, T. Nishikawa, and H. Saruwatari, "Time-domain blind source separation of non-stationary convolved signals with utilization of geometric beamforming," in Proc. *Int. Workshop on Neural Networks for Signal Processing (NNSP),* Martigny, Switzerland, 2002, pp. 445-454.

[26] H. Buchner, R. Aichner, and W. Kellermann, "A generalization of a class of blind source separation algorithms for convolutive mixtures," *Proc. IEEE Int. Symposium on Independent Component Analysis and Blind Signal Separation (ICA),* Nara, Japan, Apr. 2003, pp. 945-950.

[27] H. Buchner, R. Aichner, and W. Kellermann, "Blind Source Separation for Convolutive Mixtures Exploiting Nongaussianity, Nonwhiteness, and Nonstationarity," *Proc. Int. Workshop on Acoustic Echo and Noise Control (IWAENC),* Kyoto, Japan, September 2003.

[28] R. Aichner, H. Buchner, S. Araki, and S. Makino, "On-line time-domain blind source separation of nonstationary convolved signals," *Proc. IEEE Int. Symposium on Independent Component Analysis and Blind Signal Separation (ICA),* Nara, Japan, Apr. 2003, pp. 987-992.

[29] E. Moulines, O. Ait Amrane, and Y. Grenier, "The generalized multidelay adaptive filter: structure and convergence analysis," *IEEE Trans. Signal Processing,* vol. 43, pp. 14-28, Jan. 1995.

[30] H. Brehm and W. Stammler, "Description and generation of spherically invariant speech-model signals," *Signal Processing* vol. 12, pp. 119-141, 1987.

[31] H. Buchner, J. Benesty, and W. Kellermann, "Multichannel Frequency-Domain Adaptive Algorithms with Application to Acoustic Echo Cancellation," in J.Benesty and Y.Huang (eds.), *Adaptive signal processing: Application to real-world problems,* Springer-Verlag, Berlin/Heidelberg, Jan. 2003.

[32] H. Gish and D. Cochran, "Generalized Coherence," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP),* New York, NY, USA, 1988, pp. 2745-2748.

[33] H.H. Yang and S. Amari, "Adaptive online learning algorithms for blind separation: maximum entropy and minimum mutual information," *Neural Computation,* vol. 9, pp. 1457-1482, 1997.

[34] T.M. Cover and J.A. Thomas, *Elements of Information Theory,* Wiley & Sons, New York, 1991.

[35] S. Haykin, *Adaptive Filter Theory,* 3rd ed., Prentice Hall., Englewood Cliffs, NJ, 1996.

[36] D.H. Brandwood, "A complex gradient operator and its application in adaptive array theory," *Proc. IEE,* vol. 130, Pts. F and H, pp. 11-16, Feb. 1983.

[37] A. Papoulis, *Probability, Random Variables, and Stochastic Processes,* 3rd ed., McGraw-Hill, New York, 1991.

[38] F.D. Neeser and J.L. Massey, "Proper Complex Random Processes with Applications to Information Theory," *IEEE Trans. on Information Theory,* vol. 39, no. 4, pp. 1293-1302, July 1993.

[39] S. Amari, "Natural gradient works efficiently in learning," *Neural Computation,* vol. 10, pp. 251-276, 1998.

[40] J.D. Markel and A.H. Gray, *Linear Prediction of Speech,* Springer-Verlag, Berlin, 1976.

[41] J.W. Brewer, "Kronecker Products and Matrix Calculus in System Theory," *IEEE Trans. Circuits and Systems,* vol. 25, no. 9, pp. 772-781, Sep. 1978.

[42] M.Z. Ikram and D.R. Morgan, "Exploring permutation inconsistency in blind separation of speech signals in a reverberant environment," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP),* Istanbul, Turkey, June 2000, vol. 2, pp. 1041-1044.

[43] H. Sawada, R. Mukai, S. Araki, and S. Makino, "Robust and precise method for solving the permutation problem of frequency-domain blind source separation," *Proc. IEEE Int. Symposium on Independent Component Analysis and Blind Signal Separation (ICA),* Nara, Japan, Apr. 2003, pp. 505-510.

[44] J.-S. Soo and K.K. Pang, "Multidelay block frequency domain adaptive filter," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-38, pp. 373-376, Feb. 1990.

[45] P.C.W. Sommen, P.J. Van Gerwen, H.J. Kotmans, and A.J.E.M. Janssen, "Convergence analysis of a frequency-domain adaptive filter with exponential power averaging and generalized window function," *IEEE Trans. Circuits and Systems,* vol. 34, no. 7, pp. 788-798, July 1987.

[46] J. Benesty, A. Gilloire, and Y. Grenier, "A frequency-domain stereophonic acoustic echo canceller exploiting the coherence between the channels," *J. Acoust. Soc. Am.,* vol. 106, pp. L30-L35, Sept. 1999.

[47] G. Enzner and P. Vary, "A soft-partitioned frequency-domain adaptive filter for acoustic echo cancellation," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Hong Kong, China, April 2003, vol. 5, pp. 393-396.

[48] R.M.M. Derkx, G.P.M. Egelmeers, and P.C.W. Sommen, "New constraining method for partitioned block frequency-domain adaptive filters," *IEEE Trans. Signal Processing,* vol. 50, no. 9, pp. 2177-2186, Sept. 2002.

[49] F.J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier transform," *Proc. IEEE,* vol. 66, pp. 51-83, Jan. 1978.

[50] S. Sawada, R. Mukai, S. de la Kethulle de Ryhove, S. Araki, and S. Makino, "Spectral smoothing for frequency-domain blind source separation," in *Proc. Int. Workshop on Acoustic Echo and Noise Control (IWAENC)*, Kyoto, Japan, Sep. 2003, pp. 311-314.

[51] R. Mukai, H. Sawada, S. Araki, and S. Makino, "Real-Time Blind Source Separation for Moving Speakers using Blockwise ICA and Residual Crosstalk Subtraction," *Proc. IEEE Int. Symposium on Independent Component Analysis and Blind Signal Separation (ICA)*, Nara, Japan, Apr. 2003, pp. 975-980.

[52] J.S. Garofolo et al., “TIMIT acoustic-phonetic continuous speech corpus,” National Institute of Standards and Technology, 1993.

This page intentionally left blank

IV

AUDIO CODING AND REALISTIC SOUND STAGE REPRODUCTION

This page intentionally left blank

Chapter 11

AUDIO CODING

Gerald Schuler
Fraunhofer AEMT
schuller@emt.iis.fhg.de

Abstract In this chapter, the principles of audio coding will be described, with emphasis on low delay audio coding. Audio coding is based on psycho-acoustic masking effects, as computed by psycho-acoustic models. To use the masking effects and to obtain a good compression ratio, filter banks are used. The principles of psycho-acoustics and of the design of filter banks are presented. Further a new low delay audio coding scheme based on prediction is shown.

Keywords: Psycho-Acoustics, Filter Banks, Polyphase, Low Delay Audio Coding, Prediction

1. INTRODUCTION

Communications applications today use speech coders and usually close talking microphones and only one talker at a time and little background noise. Examples are telephones and cell phones. Speech coders have the important advantage that they have a sufficiently low encoding/decoding delay. To obtain a smooth flow of conversation it was shown that a round-trip delay of less than 90 ms is desirable [2]. This includes encoding/decoding twice (once in each direction) and the transmission delay, which can be a few 10 ms. Hence the encoding/decoding delay should be clearly less than 45 ms.

New communications scenarios include teleconferencing with many talkers and distant microphones, wireless microphones, wireless speakers, digital feedback channels for musicians, or musicians playing together remotely. These communications applications are not restricted to one talker at a time, nor to close talking microphones, and not even to speech, e.g. in the case of musicians.

What are the requirements for these applications? First the encoding/decoding delay becomes even more restrictive. If musicians play together remotely the delay should not be bigger than the delay the sound on a big stage. Similarly if there is a mix of direct sound and the encoded/decoded signal, for example in a feedback channel for musicians, in live events, or in combinations of wired and wireless speakers. In these applications the encoding/decoding delay should be less than about 6 ms.

The next requirement is for the sound quality of the decoded signal. For speech signals big changes in the signal are accepted as long as it is intelligible (telephone quality). For music such degradations are not accepted, as can be seen for example in AM Radio. There the transmission bandwidth is notably reduced, and hence the programming consists almost completely of speech, and only very little of music. Also when there is more than one talker at a time, if there is background noise or room reverberation, it helps the intelligibility to have a higher audio quality.

What is the principle of speech and audio coders? Speech coders are mainly based on a model of speech production. Only a few assumptions about the receiver are used to construct speech coders. In fact, the receiver does not even need to be the human ear, it can be a machine, like a speech recognizer. This is applied for example for speech recognition over cell phones. The problem for our newer applications is obvious, the source model does not work for non-speech signals like music or background noise.

2. PSYCHO-ACOUSTICS

To overcome the limitations of a source model, audio coding uses knowledge, or a model, of the receiver, the human ear, to obtain data compression. Psychoacoustics is a field that investigates the properties of human hearing.

The effects that are mainly used in audio coding are masking across frequency and across time. The cochlea of the inner ear is conducting a frequency decomposition, analog to the filter banks used in audio coding. This is the underlying cause of many different masking effects found in psycho-acoustics.

First simultaneous masking: if there is a signal at a certain frequency and with a certain level, a signal at the same frequency but with a certain lower level is not detected by the ear. It is “masked” by the stronger signal. The level differences for simultaneous masking depend on the level and if the signals are tones or noise like signals. A simple estimation for the simultaneous masking, which can be used for psycho-acoustic models, is:

$$O(z)/\mathrm{dB} = \alpha(14.5 + z) + (1 - \alpha)5.5,$$

where z is the signal frequency in Bark (as defined below), and α is a tonality measure, which is equal to 1 for pure tonal signals and equal to 0 for pure noise like signals [12, 13]. In general,

$$\alpha = \min(\mathrm{SFM}/\mathrm{SFM}_{\mathrm{max}}, 1),$$

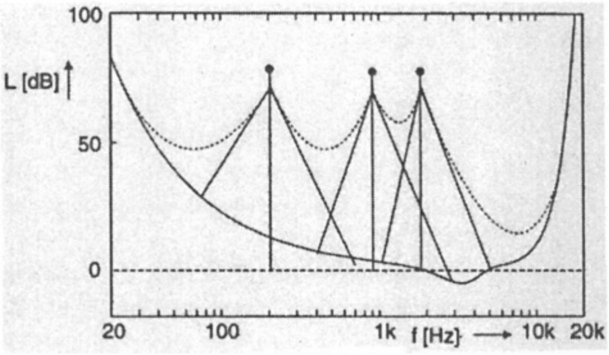


Figure 11.1 Masking threshold as superposition of individual tonal maskers and the threshold in quiet (from [12]).

with $\mathbf{SFM_{max}} = -\mathbf{60dB}$. The Spectral Flatness Measure SFM is [12, 13]

$$\mathrm{SFM} = 10\log_{10}\left(\frac{(\prod_{k=0}^{M-1} y_k^2)^{1/M}}{\frac{1}{M}\sum_{k=0}^{M-1} y_k^2}\right).$$

A weaker signal is not only masked if it is at the same frequency, but also at nearby frequencies. The level at which signals at nearby frequencies are masked is given by the "spreading function," which is a decaying function towards higher frequency differences.

The increase from lower frequencies is about 27 dB per Bark. The decrease towards higher frequencies is about 27 – 0.37 $\max(L_M - 40, 0)$ **dB** per Bark [14], where L_M is the maskers sound pressure level.

Figure 11.1 shows spreading functions of tonal maskers and their superposition together with the masking threshold in quiet, to result in the overall masking threshold for the signal.

The Bark scale is a nonlinear scale that describes the nonlinear, almost logarithmic processing in the ear. The Bark scale (z) can be approximated with the following equation:

$$z = 13 \cdot \mathrm{atan}(0.76 \cdot \mathrm{f}) + 3.5 \cdot \mathrm{atan}((\mathrm{f}/7.5)^2)$$

for z in Bark and f in kHz.

Masking is not only appearing over frequency, but also over time. If there is a stronger onset or a click at a time, a smaller click with a certain lower level a short time after it is masked. The level of the masked signal depends on the time, and is again a decaying function, towards later times. The ear needs a "recovery time" from the stronger click or onset. This is the forward masking The masking even extents to just before the click. There are processing times

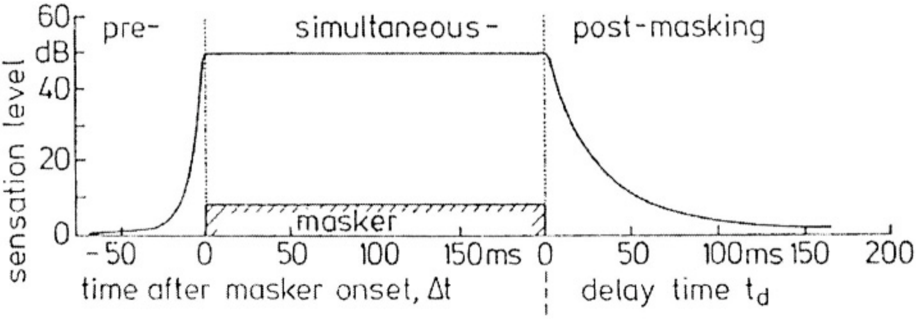


Figure 11.2 Temporal masking threshold (from [15]).

in the ear which result in a “backward masking.” This means signals just before the click are masked. But the decay of the masking threshold before the click is much faster (it practically disappears within a few milli-seconds before the click) than after it. This can be seen in Fig. 11.2.

The superposition or addition of different masking thresholds leads to some interesting effects. Consider the case where two maskers are close together in frequency. At the point where the two masking thresholds have equal intensity, they lead to an increase of the masking threshold of about 3 dB, as would be expected by the addition of their intensities. But if these maskers are further apart (more than about a critical band), the increase of masking is more, it can be up to about 12 dB [15]. This is also sketched in Fig. 11.1.

3. FILTER BANKS

An important part of most audio coders are filter banks. Early audio coders had simple Discrete Cosine Transforms of consecutive blocks of the audio signal as a filter bank, to obtain a time/frequency description. But it was found that simple block transforms are not sufficient for high quality audio coding. They lead to so-called blocking artifacts. Figure 11.3 shows the block diagram of a filter bank with critical sampling, which means the downsampling rate in each subband is equal to the number of subbands *N*. This ensures that the filter bank does not introduce any additional redundancy to the signal. The filter bank also has the so-called perfect reconstruction property, which means the subband signals can be used to perfectly reconstruct the original signal by the synthesis filter bank, but with a system delay of n_d samples.

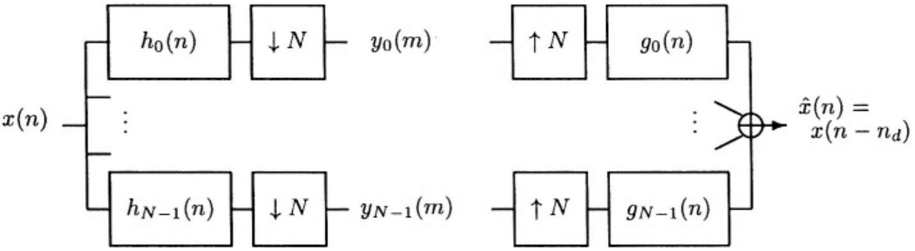


Figure 11.3 An N - channel filter bank with critical downsampling, perfect reconstruction, and a system delay of n_d samples.

3.1 POLYPHASE FORMULATION

To obtain a fast implementation, and also a mathematical description to make it easier to obtain perfect reconstruction, the polyphase description or structure is used. The effect of downsampling and upsampling in the analysis and synthesis filter bank, respectively, can be viewed as processing the signal in blocks of length *N*. The input signal is represented by an N-dimensional vector $\mathbf{x}(m)$ composed of sequences of the downsampled $x(n)$,

$$\mathbf{x}(m) := [x_0(m), \ldots, x_{N-1}(m)],$$

with

$$x_i(m) := x(mN + i),$$

and the vector of the *N* outputs of the analysis filter bank is (see also Fig. 11.3)

$$\mathbf{y}(m) = [y_0(m), \ldots, y_{N-1}(m)].$$

The z-transform of $\mathbf{x}(m)$ is given by

$$\mathbf{X}(z) = [X_0(z), \ldots, X_{M-1}(z)],$$

similar for $\mathbf{y}(m)$. The polyphase matrix for the analysis filter bank is $\mathbf{P_a}(z)$ [16],

$$\mathbf{P_a}(z) = \begin{bmatrix} P_{0,0}(z) & P_{0,1}(z) & \ldots & P_{0,N-1}(z) \\ P_{1,0}(z) & P_{1,1}(z) & & \\ \vdots & & \ddots & \\ P_{N-1,0}(z) & & & P_{N-1,N-1}(z) \end{bmatrix},$$

which contains different phases of the downsampled impulse responses,

$$P_{n,k}(z) = \sum_{m=0}^{L-1} h_k(mN + N - 1 - n)z^{-m},$$

where L is the filter length in blocks of N samples. The polyphase matrix for the synthesis filter bank is defined similarly as

$$[\mathbf{P}_{\mathrm{s}}(z)]_{k,n} := \sum_{m=0}^{L-1} g_k(mN+n)z^{-m}.$$

The analysis filtering and downsampling and the synthesis filtering and upsampling operation can then be written as

$$\mathbf{Y}(z) = \mathbf{X}(z) \cdot \mathbf{P}_{\mathrm{a}}(z)\ ,\quad \hat{\mathbf{X}}(z) = \mathbf{Y}(z) \cdot \mathbf{P}_{\mathrm{s}}(z).$$

Note that the signal vectors are multiplied from the left side, which has the advantage that it matches the signal flow in block diagrams (from left to right, see also Fig. 11.4). This is useful when converting the product into a filter structure. It can be seen that perfect reconstruction results if [16]

$$\mathbf{P}_{\mathrm{a}}(z) \cdot \mathbf{P}_{\mathrm{s}}(z) = z^{-d} \cdot \mathbf{S}^{n_t}(z), \tag{11.1}$$

where

$$\mathbf{S}(z) := \begin{bmatrix} 0 & 0 & \cdots & 0 & z \\ 1 & 0 & \cdots & & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}$$

is a “shift matrix.” The multiplication with $\mathbf{S}^{n_t}(z)$ means a shift of n_t samples in a block. Here it means the reduction of the delay of d blocks of length N by n_t samples. Fig. 11.4 shows that for $\mathbf{P}_{\mathrm{a}}(z) \cdot \mathbf{P}_{\mathrm{s}}(z) = \mathbf{I}$ the system delay of the filter bank is $N - 1$ samples. This is the blocking delay, which results from forming input blocks of length N before processing them. In general the system delay n_d is the blocking delay plus the delay introduced by the above matrices,

$$n_d = N - 1 + d \cdot N - n_t,$$

where n_t can be used for the “fine tuning” of the system delay. Because this is a general formulation for filter banks with a downsampling rate of N, it also shows that the minimum possible system delay for these filter banks is $N - 1$ samples. This is the system delay e.g. if the filter bank is a block transform of size $N \times N$.

3.2 MODULATED FILTER BANKS

Filter banks in audio coding usually have a large number of subbands. MPEG-2/4 AAC for instance has 1024 subbands, and each filter has a length of 2048 taps. For this reason it is important to have an efficient implementation,

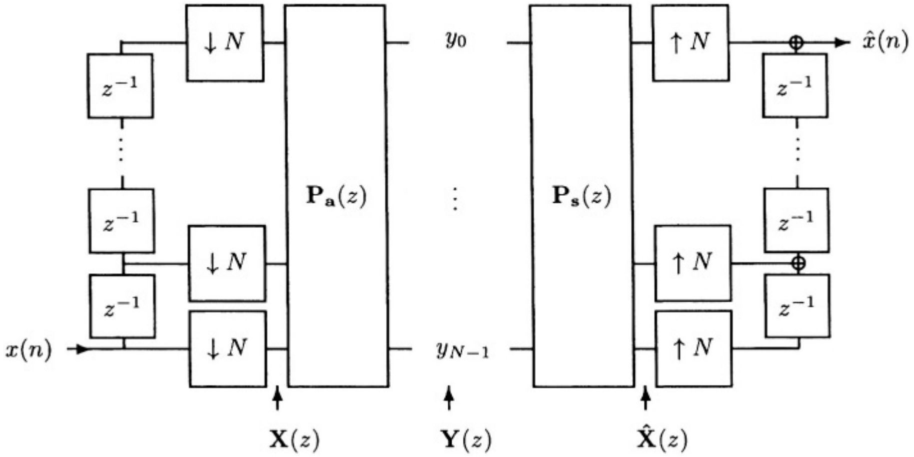


Figure 11.4 Polyphase representation of an N - channel filter bank with critical downsampling.

which can be obtained by using so-called modulated filter banks. In modulated filter banks, each filter is the product of a so-called window function (typically a low-pass filter) and a modulating function, which shifts the pass band to the center frequency of the subband. In audio coding this modulating function is typically a cosine function.

Now consider impulse responses of cosine modulated filter banks of the form

$$\begin{aligned} h_k(n) &= h(n) \cdot \cos\left(\frac{\pi}{N}(k+0.5)(n+0.5+n_a)\right), && (11.2)\\ g_k(n) &= h'(n) \cdot \frac{2}{N} \cdot \cos\left(\frac{\pi}{N}(k+0.5)(n+0.5-N+n_s)\right), && (11.3)\end{aligned}$$

where $k = 0, \ldots, N-1$, $n = 0, \ldots, LN-1$, $h(n)$ and $h'(n)$ are the analysis and synthesis baseband prototype filters or window functions. The factor 2/*N* and the additional shift of *N* in g_k is introduced to simplify the following notation. Two parameters n_a and n_s are limited to $0 \leq n_a, n_s \leq N$. This particular type of modulating function was chosen because it leads to filters with narrow passbands and high stopband attenuation.

A simple general design method for modulated filter banks is given in the following. The polyphase matrices are constructed as a product or a cascade of simpler matrices [17, 18].

The polyphase matrices $\mathbf{P}_a(z)$ and $\mathbf{P}_s(z)$ for cosine modulated filter banks as in (11.2, 11.3) have an interesting inherent structure. They can be written as a product of a sparse "filter matrix" $\mathbf{F_a}(z)$ or $\mathbf{F_s}(z)$ respectively, containing the prototype polyphase components only on the main diagonal and anti-diagonal (and is zero otherwise), a transform matrix **T**, which is a simple DCT type 4,

$$[\mathbf{T}]_{n,k} := \cos(\frac{\pi}{N}(k+0.5)(n+0.5)), \ \ 0 \le n,k < N,$$

and the shift matrices $\mathbf{S}^{n_a}$ and $\mathbf{S}^{n_s}$, adjusting the system delay. The resulting product is

$$\begin{aligned} \mathbf{P}_a(z) &= \mathbf{S}^{n_a}(z) \cdot \mathbf{F}_\mathbf{a}(z)\mathbf{T}, & (11.4) \\ \mathbf{P}_s(z) &= \mathbf{T}^{-1}\mathbf{F}_\mathbf{s}(z) \cdot \mathbf{S}^{n_s}(z). & (11.5) \end{aligned}$$

With the help of the following matrices, the matrix $\mathbf{F}_\mathbf{a}(z)$ or $\mathbf{F}_\mathbf{s}(z)$ can be written as a product of factors [19, 17]. The factors are build with the following matrices

$$\mathbf{G}_i = \begin{bmatrix} & & & & g_0^i \\ & & & \iddots & \\ & & g_{N/2-1}^i & & \\ & 0 & & & \\ \iddots & & & & \\ 0 & & & & \end{bmatrix} \qquad (11.6)$$

with zeroes in all unspecified entries, or

$$\mathbf{G}_i = \begin{bmatrix} & & & & 0 \\ & & & \iddots & \\ & & & 0 & \\ & & g_{N/2-1}^i & & \\ & \iddots & & & \\ g_0^i & & & & \end{bmatrix} \qquad (11.7)$$

in a way that both types alternate, for instance the first type for the odd indices i, and the second for the even indices. Observe that $\mathbf{G}_i^2 = 0$, meaning it is nilpotent of order 2. With the help of these matrices, the matrix $\mathbf{F}_\mathbf{a}(z)$ can be written as the product of "zero-delay matrices"$(\mathbf{I} + \mathbf{G}_i z^{-1})$, "maximum-delay matrices" $(\mathbf{I}z^{-1} + \mathbf{G}_i)$, and a diagonal coefficient matrix $\mathbf{D} = \text{diag}(d_0, \ldots, d_{N-1})$,

$$\mathbf{F}_\mathbf{a}(z) = \prod_{i=0}^{\mu-1}(\mathbf{I}z^{-1} + \mathbf{G}_i) \cdot \prod_{j=0}^{\nu-1}(\mathbf{I} + \mathbf{G}_{\mu+j}z^{-1}) \cdot \mathbf{D}. \qquad (11.8)$$

Observe that these matrix factors can also be seen as so-called Lifting steps, as seen in Fig. 11.5.

To obtain the synthesis polyphase matrix $\mathbf{P}_\mathbf{s}(z)$ for perfect reconstruction, the inverse is needed. The inverse of the factors is simply

$$\begin{aligned} (\mathbf{I} + \mathbf{G}_i z^{-1})^{-1} &= (\mathbf{I} - \mathbf{G}_i z^{-1}), \\ (\mathbf{I}z^{-1} + \mathbf{G}_i)^{-1} \cdot z^{-2} &= (\mathbf{I}z^{-1} - \mathbf{G}_i). \end{aligned}$$

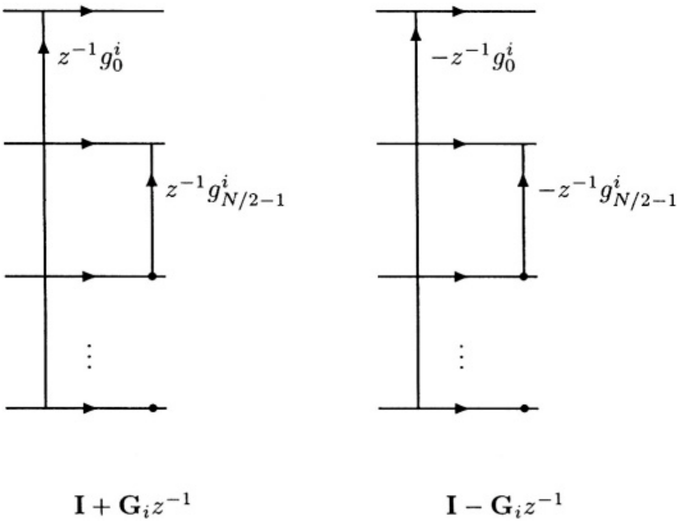

$z^{-1}g_0^i$
$z^{-1}g_{N/2-1}^i$
$-z^{-1}g_0^i$
$-z^{-1}g_{N/2-1}^i$
$\mathbf{I} + \mathbf{G}_i z^{-1}$
$\mathbf{I} - \mathbf{G}_i z^{-1}$

Figure 11.5 The flow graph corresponding to the factors $\mathbf{I} + \mathbf{G}_i z^{-1}$ and $\mathbf{I} - \mathbf{G}_i z^{-1}$ shows the lifting-like structure.

In such a way, the synthesis filter matrix $\mathbf{F_s}(z)$ for perfect reconstruction becomes

$$\mathbf{F_s}(z) = \mathbf{D}^{-1} \prod_{\nu-1}^{j=0} (\mathbf{I} - \mathbf{G}_{\mu+j} z^{-1}) \prod_{\mu-1}^{i=0} (\mathbf{I} z^{-1} - \mathbf{G}_i). \tag{11.9}$$

With this synthesis filter matrix, the concatenation of analysis and synthesis filter bank, or the product of analysis and synthesis polyphase matrices is

$$\mathbf{P_a}(z)\mathbf{P_s}(z) = z^{-2\mu} \cdot \mathbf{S}^{n_a+n_s}(z),$$

which means we have perfect reconstruction with a system delay of $2\mu N - n_a - n_s$ samples.

A simple and widely used example for a cosine modulated filter bank is the so-called MDCT (short for "modified discrete cosine transform"). It has $n_a = n_s = N/2$, $\nu = \mu = 1$, a filter length L = 2N, and $h(n) = h'(n)$. The resulting analysis filter matrix is

$$\mathbf{S}^{N/2}(z)\mathbf{F_a}(z) = \begin{bmatrix} 0 & h(0)z^{-1} & h(N) & 0 \\ & \iddots & \ddots & \\ h\left(\frac{N-2}{2}\right)z^{-1} & 0 & & h\left(\frac{3N-2}{2}\right) \\ h\left(\frac{N}{2}\right)z^{-1} & & & -h\left(\frac{3N}{2}\right) \\ & \ddots & \iddots & \\ 0 & h(N-1)z^{-1} & -h(2N-1) & 0 \end{bmatrix}. \tag{11.10}$$

This matrix includes the multiplication with the shift matrix, hence it has this diamond like shape. The synthesis filter matrix is

$$\mathbf{F_s}(z)\mathbf{S}^{N/2}(z) = \begin{bmatrix} & h'\left(\frac{N-2}{2}\right) & h'\left(\frac{N}{2}\right) & \\ & \iddots & \ddots & \\ h'(0) & & & h'(N-1) \\ h'(N)z^{-1} & & & -h'(2N-1)z^{-1} \\ & \ddots & \iddots & \\ & h'\left(\frac{3N-2}{2}\right)z^{-1} & -h'\left(\frac{3N}{2}\right)z^{-1} & \end{bmatrix}. \tag{11.11}$$

We have perfect reconstruction if $\mathbf{S}^{N/2}(z)\mathbf{F_a}(z)\mathbf{F_s}(z)\mathbf{S}^{N/2}(z) = \mathbf{I}\cdot z^{-1}$. Matrix algebra shows that this invertibility is obtained with

$$\begin{aligned} h'(n) &= \frac{h(n)}{h(n)h(2N-1-n)+h(N+n)h(N-1-n)}, \\ h'(N+n) &= \frac{h(N+n)}{h(n)h(2N-1-n)+h(N+n)h(N-1-n)}. \end{aligned}$$

It can be seen that the denominator is the determinant of coefficients of 2 × 2 subsets of matrix (11.10). If this determinant is equal to one, $h'(n) = h(n)$. A simple window function which fulfills this condition is the sine window

$$h(n) = \sin(\frac{\pi}{2N}(n+0.5))$$

for $n = 0, \ldots, 2N-1$. This is already a commonly used window for audio coding. Another window is the so-called Kaiser-Bessel Derived (KBD) window, which is numerically optimized for a higher stopband attenuation, and used for instance in the MPEG-AAC and Dolby AC-3 coders [20].

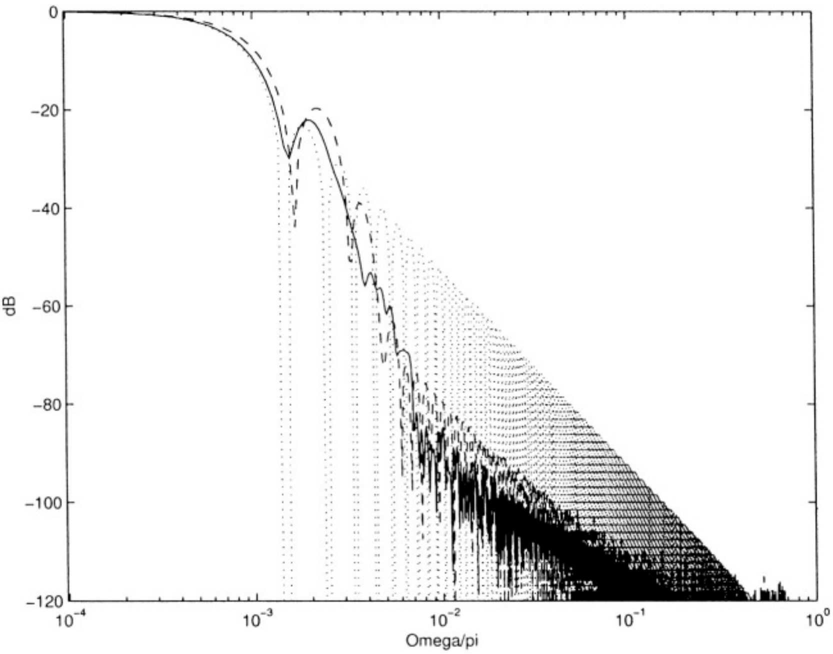


Figure 11.6 **Frequency responses of the sine window (dotted), the KBD window (dashed), and the low delay window (solid line).**

Both windows lead to so-called orthogonal filter banks (the impulse responses of analysis and synthesis are time reversed versions of each other), with "unitary" polyphase matrices, i.e. $\mathbf{P_a}^{-1}(z) = \mathbf{P_a}'(z^{-1})$. In this case the system delay equals the length of the filters minus one, $n_d = LN - 1$. In the general case, with the above described design method, it is possible to obtain low delay filter banks with a lower delay than the length of the filters. This can be used for instance to obtain improved frequency responses without the need to increase the system delay. For example the frequency responses of the sine and the KBD windows can be surpassed by using a filter bank with twice the filter length but still the same delay. This can be seen in Fig. 11.6. Here the frequency responses of three different window functions for 1024 band filter banks are compared. The sine and KBD windows have lengths of 2048 taps and the corresponding filter banks have a system delay of 2047 samples. The low delay window has a length of 4096 taps, but a system delay of only the usual 2047 samples. Because of the higher length, it has a better far off attenuation than the KBD window and a comparable nearby attenuation as the sine window. The corresponding window functions can be seen in Fig. 11.7.

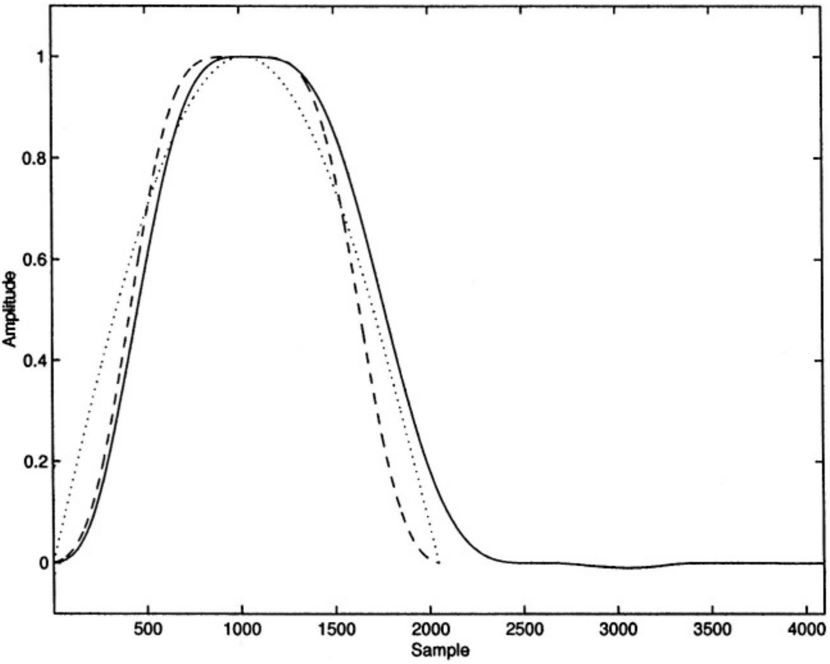


Figure 11.7 The sine window (dotted), the KBD window (dashed), and the low delay window (solid line).

3.3 BLOCK SWITCHING

Most audio coders use block switching to avoid pre-echo artifacts. Block switching means that the filter bank is switched to a different number of subbands and a different filter length, all while processing the signal. This is done while maintaining perfect reconstruction at all times. How it is done for the sine window can be seen in Fig. 11.8. In (11.10) it can be seen that the first and last *N/2* elements of the window change direction, they reverse the time. The operation of multiplicating the signal with (11.10) can also be seen as "time-domain aliasing," the first and last *N/2* elements of the signal are "folded" onto a sequence of length *N*. In order to invert this time-domain aliasing or folding we need the inverse structure as in (11.11). When we switch to a shorter window, only a shorter length of the time domain aliasing can be reversed. Hence we need a switching window which has a shorter length of the time-reversal in the end, so that it can be inverted by the following short window. Usually the switch-down window contains the first half of the original long window, then a sequence of $N_l/2 - N_s/2$ ones (with N_l the number of subbands for the long window, and N_s for the short window), followed by the last half of the short window. In order to keep the same overall blocking structure, this switch-down window is followed by a multiple of N_l/N_s short windows, and then a switch-up window is used. The latter is a time reversed version of the switch-down window. The sequence of a switch-down window for 1024 bands, 8 short windows for 128 bands, and a switch-up window can be seen in Fig. 11.8.

Low delay filter banks can be made time-varying by using time-varying coefficients in the matrices (11.6),(11.7), to obtain $\mathbf{G}_i(m)$, with the time index m, and also a time-varying diagonal coefficient matrix $\mathbf{D}(m)$. Observe a special commutation rule for these time-varying systems. If the signal is multiplied from the left, then [17]

$$\mathbf{G}_i(m) \cdot z^{-1} = \cdot z^{-1}\mathbf{G}_i(m-1). \tag{11.12}$$

This means a signal which first passes $\mathbf{G}_i(m)$ and is then delayed by one block is equal to a signal which is first delayed and the passes $\mathbf{G}_i(m-1)$, with equally delayed coefficients. Hence for the maximum delay matrices in (11.8), (11.9) the following is true

$$(\mathbf{I}z^{-1} + \mathbf{G}_i(m)) \cdot (\mathbf{I}z^{-1} - \mathbf{G}_i(m-1)) = \mathbf{I}z^{-2}.$$

Since the maximum delay matrices together with their inverses introduce a delay, (11.12) also means that the following inverses have to have accordingly delayed coefficients. In this manner also the number if bands can be switched. A suitable number of coefficients has to be set to zero, and for the transform matrix a smaller DCT is used, padded with zeros [17].

4. CURRENT AND BASIC CODER STRUCTURES

Most audio coders consist of a few main building blocks (Fig. 11.9). The filter bank, which typically has 1024 bands, and which can be switched to 128 bands, divides the audio signal into spectral components. In general more subbands lead to higher coding gain, which means a lower bit-rate. But more subbands also mean longer impulse responses of the filter bank, and this can lead to pre-echo artifacts at transients, like attacks [1]. Hence it can be switched to a lower number of subbands with shorter impulse responses to avoid pre-echoes at transient parts of a signal. For the switching decision a “look ahead” of one block is needed, which introduces a delay. The psychoacoustic model is mostly proprietary. The quantization step size in each subband is controlled by the psycho-acoustic model. This operation introduces the quantization noise. The quantization step-size has the be transmitted to the decoder for the decoding process. Hence this information has to be parameterized, in order to minimize the required bit-rate of this side-information. This parameterization determines the noise shape and hence should approximate the masking threshold as closely as possible. The usual approach is to use a piece-wise constant function. This is appropriate because the masking threshold is usually a smooth function over

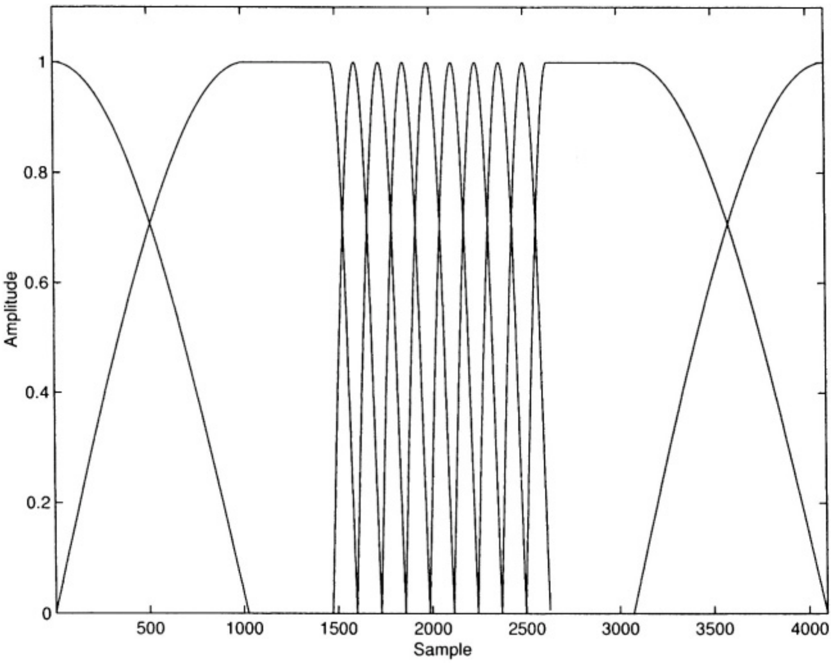


Figure 11.8 Diagram of a sequence of switching windows. Left the switch-down window, then 8 short windows, then the switch-up window.

frequency. A frequency section with the same quantizer step size is often called a "scale factor band." The AACcoder, for instance, has 49 scale factor bands, with increasing numbers of subbands per scale factor band towards higher frequency, roughly according to the Bark scale. After the quantization the signal is entropy coded, usually with Huffman coding, with a set of different Huffman codebooks. For each scale factor band the optimal codebook is chosen for the encoding, and its index is transmitted to the decoder, also as side information. Especially when switching to the shorter blocks this system leads to large fluctuations in instantaneous bit-rate. To smooth the bit-rate demand of the coder, buffering of the bit-stream is used.

Examples are MPEG2/4 AAC, and PAC. AAC is an abbreviation for "Advanced Audio Coding" and is the latest audio coder of the MPEG series of coders. PAC is for "Perceptual Audio Coding," and is a proprietary audio coder with the same developmental roots as the AAC.

A characteristic of the MPEG coders is that only the decoder and the bit-stream syntax is standardized. The encoder is described only as an "informative" part. This leaves room for improvements on the encoder side, and hence different encoders can have different compression performances, for instance depending on the quality of the psycho-acoustic model.

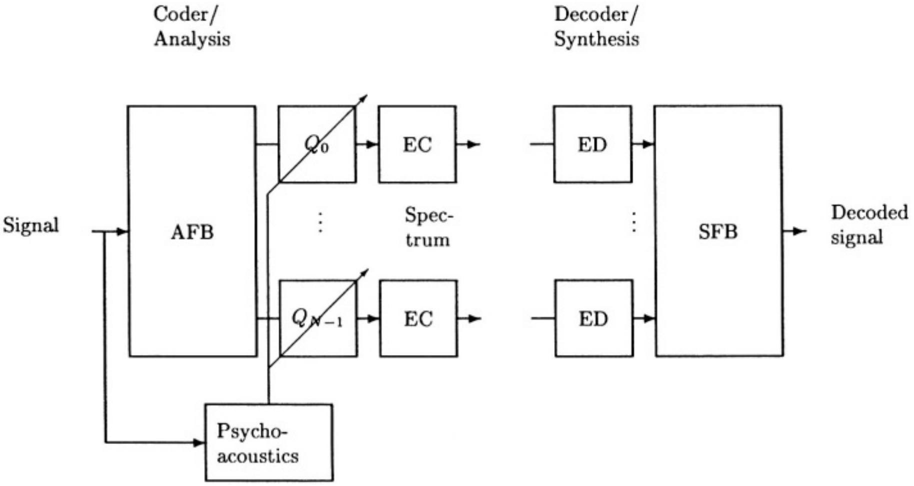


Figure 11.9 Principle of a subband based audio coder. AFB: analysis filter bank, SFB: synthesis filter bank, Q: quantizers, EC: entropy coding, ED: entropy decoding.

The desire for a high compression ratio leads to a high number of subbands. This in turn leads to long impulse responses and a high system delay of the analysis/ synthesis filter banks. Together with the look ahead delay for the switching decision and the delay introduced by buffering the bit-stream, this leads to a encoding/ decoding delay which is too high for communications purposes, usually from around 100 ms to a few hundred ms.

5. STEREO CODING

The goal of stereophonic coding is to produce a spatial acoustic impression during playback. To obtain this effect we can take a look at how the ear is determining the directions of a sound, the psycho-acoustic effects it is using. There are two main categories of clues or effects the ear is using. The first is the interaural time differences (ITD). The sound from a sound source closer to one ear reaches that ear first. From the time differences the brain can estimate the horizontal direction of the sound source. The second category is the interaural level differences (ILD). The sound from a sound source closer to one ear will have a higher level at that ear, because the sound to the other ear has to travel around the head.

If we look more closely on how the ear and the brain is using those clues, we find that for the ITD below about 800 Hz, the ear uses mostly waveform or phase information. For frequencies above 800 Hz it uses the energy envelope of the sound. The ILD are more important at higher frequencies. Since the ear uses both categories for the estimation of the sound direction, they can be

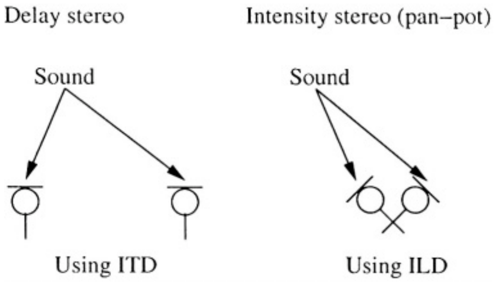


Figure 11.10 Variants of stereo recordings.

interchanged within limits for coding purposes. Figure 11.10 shows how those two effects can be used to make stereophonic recordings.

For coding a stereophonic signal the spatial distribution of the quantization noise is important. The optimal masking of the quantization noise occurs, when it has the same apparent spatial position as the signal. If the quantization noise and the signal have different spatial positions, unmasking of the quantization noise can occur, even if it was masked in the mono case. This is quite a remarkable effect. It means, if one takes two identical or very similar audio signal, encodes them with the quantization noise just at the masking threshold, but with different quantization noise (different phase), and plays those two signals as left and right channel, then the quantization noise becomes audible, even if it was not audible in each channel in itself! To avoid this unmasking of the quantization noise, the masking threshold has to be lowered. The difference of the original and lowered masking threshold is called the "binaural masking level difference" (BMLD), which is up to 18 dB.

This effect poses a problem for stereophonic coding. If the two channels, left and right, are simply encoded separately by a monophonic audio coder, and if the audio signal is centered, like a news speaker or an instrument in the middle, then unmasking of the quantization noise can occur. The decoded signal still appears in the center, whereas the uncorrelated quantization noise appears to come from the sides. A further problem is that the redundancy between the two channels is not used for compression of the signal. To solve this problem for centered signals, the signal is first processed by "matrixing," or more specifically by computing the sum and difference of the two channels. The two channels for the encoding are then the "mid" (M), which is the average of the left (L) and right (R) channel, in effect the mono signal, and the "side" (S), the difference of left and right,

$$M = (L + R)/2, \quad S = (L - R)/2.$$

The decoder inverts this operation after decoding the Mid and Side channel,

$$L = M + S, \quad R = M - S.$$

Observe that this operation is lossless in the absence of quantization. It can also be seen as a principal axis transform, where the coordinate system is rotated by 45 degrees, from a left/right to a mid/side coordinate system. If the audio signal contains mostly centered sources, most of the energy will be in the mid channel, and only very little in the Side channel. The reduced energy in the Side channel reduces the required bit-rate for the stereo encoding. Also observe the important property that now the quantization noise for the mid channel appears indeed from the center after decoding, since the mid channel appears in both, the left and right channel, after decoding. This way we solved the problem for centered signals. But what about signals from the sides? If we have an audio signal predominantly from one side, the matrixing poses a problem. It spreads the signal to the other channel, which means that unnecessary bits are needed, and it also leads to a quantization noise spread between the left and right channel. This means, for signals which are not centered it is advantageous to switch back to separate left/right coding. This switching information has to be transmitted to the decoder as side-information. Usually an audio signal cannot be clearly be categorized into centered or not. There are usually many different sound sources at different spatial locations. Fortunately they usually occupy different regions across frequency, at least as long as they are clearly distinguishable by the ear. This leads to a more efficient switching algorithm for the mid/side decision. The signal is divided into subbands, for instance roughly according to the Bark scale. The decision for Mid/Side or left/right coding is then made in each of these subband independently. This increases the necessary side information, but overall leads to a more efficient stereo coding. This is the most widely used stereo coding strategy. It is used for instance in MPEG-AAC, AC-3, and PAC.

Mid/side coding can be used to obtain high quality stereo coding, but it also leads to some overhead in bit-rate. For lower bit-rates it can be worthwhile to reduce this overhead at the expense of the precision of the spatial rendering of the audio signal. This goal can be obtained by the so called intensity stereo. Behind intensity stereo is the assumption the we have a predominantly centered audio signal. It is a lossy coding, which means we lose precision for the spatial position of the sound sources, and it uses only the ILD to obtain the spatial effects. Usually it is used above a certain cut-off frequency, for instance 4 kHz. In principle it works like mid/side coding, but instead of the side channel there is only a side-information about the amplitude in each channel. The mid channel becomes a so-called coupling channel. It is the mono version of the signal. Compared to the mid channel it contains a phase adjustment, to avoid cancellations of signals by the addition. The power in the Coupling channel

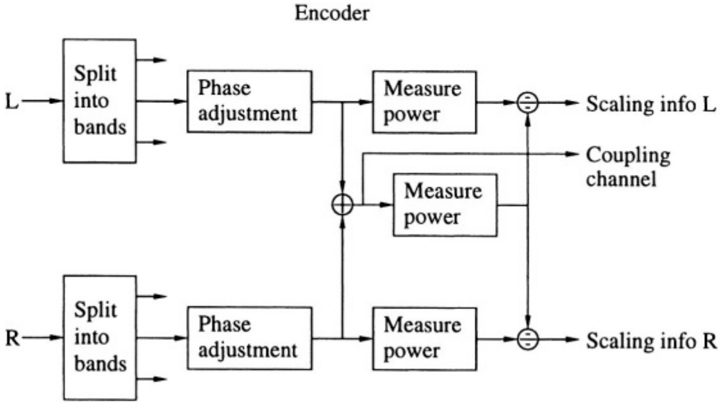


Figure 11.11 Diagram of an intensity stereo encoder.

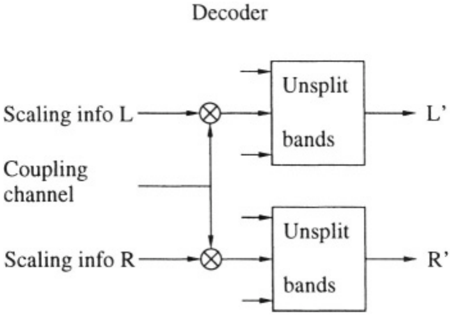


Figure 11.12 Diagram of an intensity stereo decoder.

and in the left and right channel are measured. The relation of the left and right power to the power in the coupling channel is then transmitted as scaling side information to the decoder, together with the coupling channel. This process is done in subbands, just like for mid/side coding. This can be seen in Fig. 11.11, [20]. The decoder then plays the coupling channel, but scaled for the left and right side according to the scaling information in the subbands, as in Fig. 11.12. Intensity stereo is used at the lower bit-rates in MPEG-AAC, AC-3, PAC, and MPEG1/2-Layer3 (MP3).

6. LOW DELAY AUDIO CODING

In full-duplex applications like teleconferencing, echoes bouncing back from the far-end can occur, and unnatural delays in response times in interaction between parties are to avoid. The latter may be a problem in a conversation application if the *round-trip time,* RTT, from near-end to a far-end and back

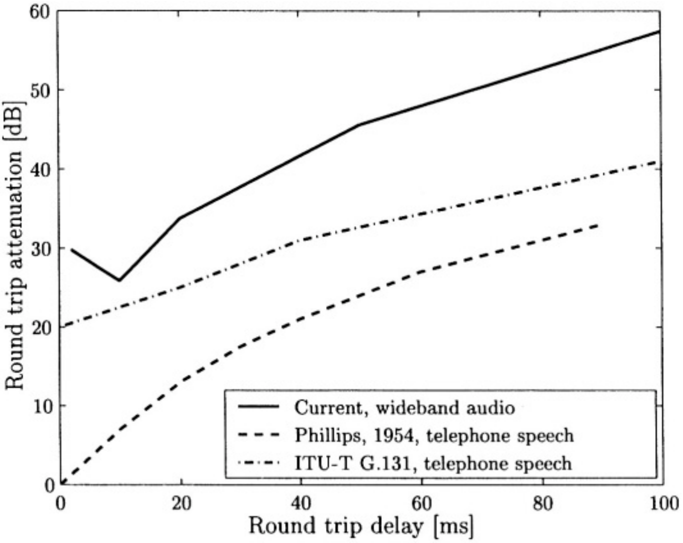


Figure 11.13 **Required attenuation in a round trip loop.**

exceeds 90 ms [2]. Reflections are not a problem if the acoustic round-trip path can be eliminated, e.g., by using a combination of headphones and close-talk microphones. Otherwise, it is necessary to use acoustic echo cancellation, AEC, to attenuate return-path reflections. A distinct reflection arriving at listeners ears 40-50 ms after the direct sound is called *echo.* At low RTTs, the perceived effect is a colorization of the signal.

Figure 11.13 shows the required attenuation for a single reflection in three different experiments. The dashed curve is from [21] and represents the round-trip attenuation at *acceptable level* for telephone speech. The dashed-dotted curve shows the corresponding ITU-T G. 131 recommendation. The solid curve in Fig. 11.13 shows measurement data averaged over widely used high-quality audio test material including *Castanets, Suzanne Vega, female and male speakers, and flute* at the sampling rate of 32 kHz [22].

The results are qualitatively in line with earlier results but, as expected, show significantly higher requirements for attenuation especially at high delays. The dip for the very low delays can be explained by the fact that very low delays (which are not realistic in most applications) result in a comb-filter like effect, which becomes noticeable.

The data suggest that a reduction of 10 ms in the round-trip delay corresponds to a 3-4 dB drop in the requirements for echo cancellation. For example, if the algorithmic coding delay is diminished from 20 to 6 ms, the requirements for AEC are down by more than 10 dB. The required high attenuation for echoes (RTT > 50 ms) is very difficult to achieve. Therefore, a 25 ms one-way delay is a realistic upper margin for echo-free audio communications. The speed of light in an optical fiber is approximately 200 km/ms. Hence, one may roughly estimate that a 1 ms decrease in algorithmic coding delay corresponds to a 100 km increase in the range of echo-free communications. This all suggest that the coding delay should be below about 10 ms which is also close to the recommendations for low-delay speech coding [3].

To obtain an audio coder for communications applications, for instance the AAC- low delay coder was developed. It has the same basic structure as the AAC coder. To obtain a lower delay it features shorter filters and fewer subbands. It has 480 subbands with filters of length 960 taps, and has no switching to a lower number of subbands to avoid the look ahead for the switching decision. Instead it has a window shape switching to a window with the same number if subbands but less overlap and shorter length, and temporal noise shaping [23] to reduce pre-echo artifacts [24], Avoiding the switching also leads to less fluctuations in the bit-rate demand, which leads to a reduced buffer size and a lower delay. Without the buffering, it has a delay of 960 samples, or 20 ms at 48 kHz sampling rate (accordingly scaled at other sampling rates). But its reduction of the number of subbands also leads to a reduced compression performance compared to the AAC coder. Its compression performance is roughly comparable to MP3, the predecessor of the AAC coder. Its operating range is usually between 48 and 96 kb/s.

The ITU G722.1 coder is also a subband coder, and it is more specialized towards speech coding. It has 320 subbands, operates at 16 kHz sampling rate and at bit-rates between 24 and 32 kb/s. It has a delay of 640 samples or 40 ms at 16 kHz sampling rate.

There is a trade-off between the delay and the compression performance with subband coding. To avoid this trade-off or connection, a different coding principle is needed. Predictive coding has the same asymptotic compression performance (for stationary signals and infinitely many subbands or infinitely long predictors) [25], but it has the advantage that a predictor does not introduce a delay, no matter how long it is. This is a principle which has long been used in speech coding, as in ADPCM coders like G.726 or G.727. Also the speech coders used in cellular phones are based on predictive coding. The problem for its application in audio coding is the application of the psycho-acoustic model. It provides the quantization step size for each frequency band, and is hence directly suited for subband coding. To obtain a comparable noise shaping with predictive coding, a suitable psycho-acoustically controlled pre- and post-filtering can be used [4, 7]. In the first stage of the encoder the pre-filter is used. It is controlled by the psycho-acoustic model and its computed masking threshold $M(f)$, such that the frequency response $H(f)$ of the pre-filter is inverse to the masking threshold,

$$H(f) = \frac{1}{|M(f)|}. \tag{11.13}$$

The filtering of the input audio signal can then be seen as normalizing the signal to its masking threshold. This has the consequence that distortions below unity in magnitude are inaudible. Hence a simple uniform stepsize quantizer is used to quantize the signal. The operation of pre-filtering and quantizing can be seen as irrelevance reduction. All inaudible components are removed. Since further distortions would be audible, this stage needs to be followed by a lossless coding stage, to obtain the final compression and bit-stream. The decoder has a lossless decoder as a first stage, followed by the post-filter. This post-filter has a frequency response which is inverse to the pre-filter. Its frequency response corresponds to the masking threshold as computed by the psycho-acoustic model in the encoder. Hence side-information is needed to transmit the filter coefficients of the pre-filter to the decoder. This can be done efficiently using so-called line spectral frequencies, as known from speech coding [26]. The psycho-acoustic model in the encoder is still based on a subband decomposition with a filter bank. But since this filter bank is not used for the redundancy reduction, it can have fewer subbands and shorter filters than in conventional audio coders. With 128 subbands for the psycho-acoustic model, the first stage in the encoder introduces a delay of about 128 samples. A block diagram of the system can be seen in Fig. 11.14.

The pre-filter is constructed like a predictor, to make it easily invertible. With a direct form FIR implementation and the order of the pre-filter of K its output $y(n)$ is related to its input $x(n)$ through

$$y(n) = x(n) - \sum_{k=1}^{K} a_k x(n-k). \tag{11.14}$$

The inverse DFT of $|M(f)|^2$ gives the auto-correlation function $r_{mm}(n)$. Then the filter coefficients a_k are obtained by solving the linear equation system [5]

$$\sum_{k=0}^{K-1} r_{mm}(|k-n|)a_k = r_{mm}(n+1), \quad 0 \leq n < K. \tag{11.15}$$

The coefficients a_k describe the masking threshold in a parametric form, just like the scale factors in conventional audio coders. Only here it is in effect an approximation by a polynomial (the frequency response of the post-filter). Compared to the conventional piece-wise constant form it has the advantage that it avoids unnatural steps in the approximation, it is inherently a smooth function and hence has the ability of a closer approximation of the masking threshold. It

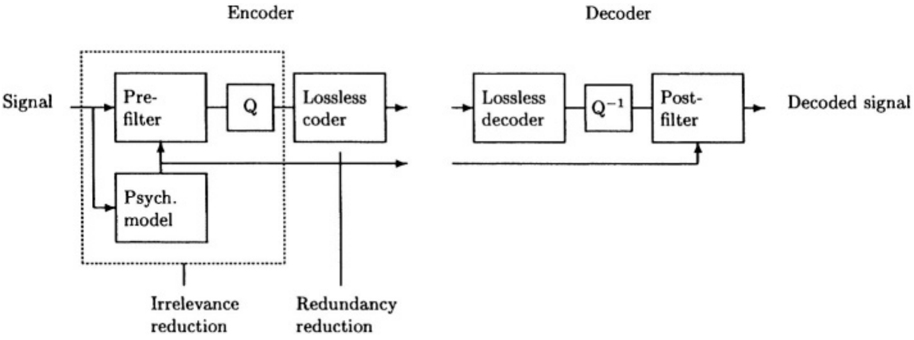


Figure 11.14 The low delay audio coding scheme using psycho-acoustic pre- and post-filters and predictive lossless compression.

was found that a 12th order filter is sufficient for this approximation. To further improve this approximation, frequency warped filters can be used in order to better follow the details of the masking threshold on the Bark scale [6].

The pre-filter and quantizer is followed by the lossless coder. Current lossless audio coders are typically based on block wise forward prediction. The prediction coefficients for a block are transmitted as overhead, and the residuals are Huffman coded and transmitted. This means there is a delay of at least one block size. To obtain a low delay, backward adaptive predictive coding using the LMS algorithm is used [27], which is also a standard technique in low-delay speech coding [3]. The LMS algorithm is also widely used in on-line automatic control, or acoustic echo cancellation (cf. [5]).

Let $x(n)$ be the signal at time n, and $\mathbf{x}^T(n)$ is defined as $\mathbf{x}^T(n) := [x(n-L+1), ..., x(n)]$ where L is the order of the prediction. An L'th-order predictor is of the form

$$P(\mathbf{x}(n-1)) = \mathbf{x}^T(n-1) \cdot \mathbf{h}(n), \qquad (11.16)$$

where $\mathbf{h}(n)$ is the L-dimensional vector of predictor coefficients at time n. $\mathbf{h}(n)$ is updated with the normalized LMS:

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \frac{e(n)}{1 + \lambda ||\mathbf{x}(n-1)||^2} \mathbf{x}(n-1). \qquad (11.17)$$

with $e(n)$ being the prediction error. This is a special case of the normalized LMS [5], i.e. with only one tuning parameter λ to trade off adaptation speed and accuracy.

To obtain a high compression ratio, cascading and soft switching between the cascaded predictors can be used [8]. The filter bank in conventional audio coding has 2 modes, one with a high number of bands (typically 1024) but reduced time resolution, and one with a lower number of bands (typically 128)

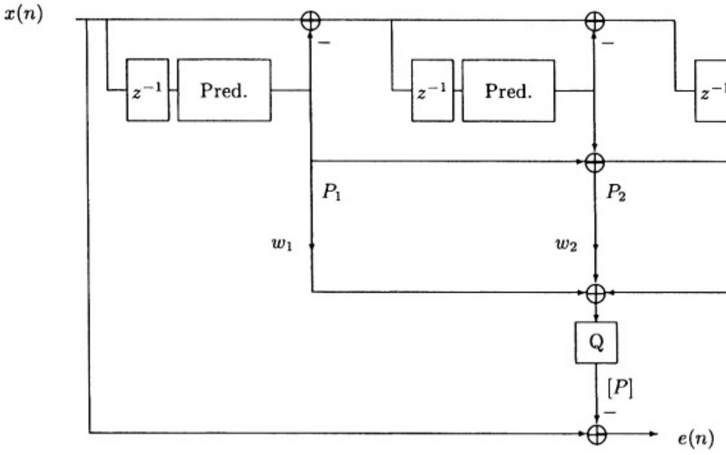


Figure 11.15 The WCLMS lossless encoder. Q denotes rounding.

but a higher time resolution. The mode depends on the signal and is hard switched (either one or the other). The analog of a filter bank with many bands in subband coding is a predictor with high order in predictive coding. In predictive coding, having more modes is simpler than in subband coding. This makes it possible to have 3 instead of 2 modes, and to obtain *soft* switching instead of hard switching. This is useful because it increases the prediction accuracy and also avoids high bit-rate peaks, as follows. Speech/audio signals have varied orders of correlations. Very non-stationary signals like sounds from castanets need a short predictor that is able to track the signal fast enough, whereas more stationary signals as sounds from flutes require higher prediction orders to accurately model the signal with all its spectral details.

The LMS prediction is applied three times in a cascade, (Cascaded Weighted LMS, or WCLMS) leading to the predictors P_1, P_2 and P_3. Since the residuals $e_1(n)$ of the first predictor are not integers but floating point numbers, they cannot be reproduced and stored in finite precision without losing accuracy. This is not a problem for a single LMS since its input $x(n)$ are integers (PCM signals). However, in the second and third stages in cascading LMS, the non-integer residuals are the inputs to improve the accuracy of their prediction. But when the encoding and decoding sides have different rounding precisions, we are not able to synchronize the two sides and the encoder and decoder will produce different outputs. This problem is solved by limiting the precision of the residuals in a defined manner, e.g. using 8 bit precision after the fractional point. A diagram of a WCLMS encoder can be seen in Fig. 11.15, and the decoder in Fig. 11.16.

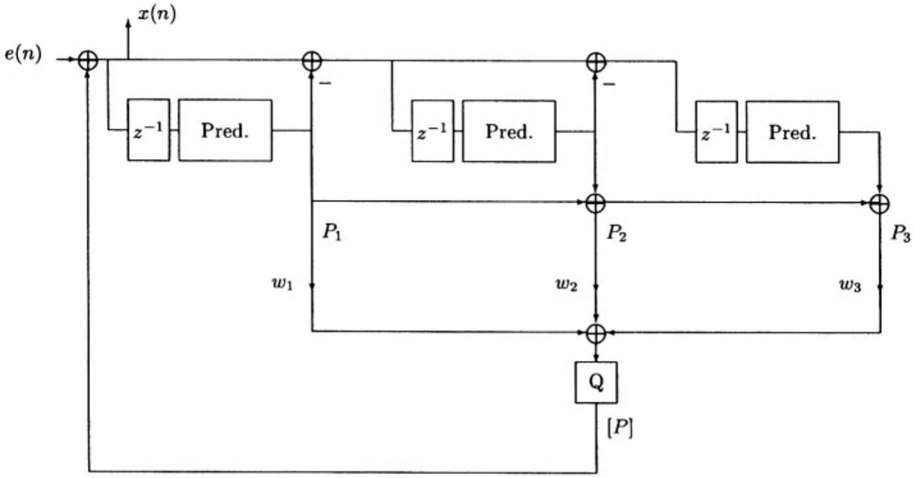


Figure 11.16 The WCLMS lossless decoder. Q denotes rounding.

By using the cascade of predictors one of the main issues is how to select or combine these predictors. Bayesian statistics uses weighted combinations for an improved prediction performance (cf. [28]). Using this approach the model-based predictors P_i can be combined into

$$\sum_i w_i P_i, \quad w_i \geq 0, \sum_i w_i = 1, \tag{11.18}$$

where w_i is the posterior (i.e. based on the observed data) probability that P_i is "correct" given data to date, which can be viewed as a measure of the goodness-of-fit of the model or predictor P_i.

For most audio signals the assumption of a Laplacian probability distribution yields a good fit to estimate the weights w_i. With a "forgetting factor" μ and the joint probability density function, this leads to the estimate of the weight w_i:

$$w_i(n) \propto e^{-c(1-\mu)\sum_{i=1}^{\infty} |e_i(n-i)| \cdot \mu^{(i-1)}}. \tag{11.19}$$

The final stage is an entropy coder for the prediction error. Again, block based approaches like block based Huffman coding cannot be used because of their inherent delay. But (backward) adaptive schemes can be used, as the adaptive Huffman coder or the Arithmetic coder.

The resulting audio coder has a delay which is mainly determined by the psycho-acoustic model in the encoder, with its block size of 128 samples. Taking into account some delay for the entropy coding, the overall delay is about 200 samples, or 6 ms at 32 kHz sampling rate. This is sufficient for the most delay critical applications. At the same time it has a compression ration which is comparable to conventional audio coders.

7. CONCLUSIONS

Irrelevance reduction is necessary for audio coding, because we cannot rely on a source model, as in speech coding. For that reason the basics of psycho-acoustic models was described. These models work mainly in the frequency domain. This makes the use of subband coding convenient for audio compression. Communications applications need a very low end-to-end delay of only a few milli-seconds. But subband coding has an inherent trade-off between compression performance and the resulting end-to-end delay. This would lead to a low compression performance for a very low delay. To avoid this trade-off, a different coding principle is useful. It is predictive coding, which does not have this inherent trade-off, and which is already widely used in speech coding. The problem is the application of psycho-acoustic models, who work in the frequency-domain. It was seen that this problem can be solved by using psycho-acoustically controlled pre- and post-filters, and that indeed audio coders can be designed using predictive coding, using this method. This type of audio coder has a delay of only a few milli-seconds, and hence can be used even in very delay critical communications applications.

References

[1] Technical Council of the AES: CD "Perceptual audio coders: what to listen for," Audio Engineering Society, New York.

[2] N. Kitawaki and K. Itoh, "Pure delay effects on speech quality in telecommunications," *IEEE J. Sel. Areas in Comm.,* vol. 9, pp. 586–593, May 1991.

[3] J.-H. Chen, R. V. Cox, Y.-C. Lin, N. Jayant, and M. J. Melchner, "A low-delay CELP coder for the CCITT 16 kb/s speech coding standard," *IEEE J. Sel Areas in Comm.,* vol. 10, pp. 830–849, June 1992.

[4] B. Edler and G. Schuller, "Audio coding using a psychoacoustic pre- and post-filter," ICASSP 2000, Istanbul, Turkey, pp. 11-881:884.

[5] S. Haykin, *Adaptive Filter Theory.* Englewood Cliffs, N.J.: Prentice Hall, 1999.

[6] A. Härmä, U. K. Laine, and M. Karjalainen, "Backward adaptive warped lattice for wideband stereo coding," in *Proc. of EUSIPCO'98,* (Greece), 1998.

[7] B. Edler, C. Faller, G. Schuller, "Perceptual Audio Coding Using a Time-Varying Linear Pre- and Post-filter," AES Symposium, Los Angeles, CA, Sept. 2000

[8] G. Schuller, B. Yu, D. Huang, "Lossless coding of audio signals using cascaded prediction," in *Proc. ICASSP,* Salt Lake City, Utah, May 2001

[9] S. Dorward, D. Huang, S. A. Savari, G. Schuller, and B. Yu, "Low Delay Perceptually Lossless Coding of Audio Signals," Data Compression Conference, Snowbird, UT, March 2001, pp. 312-320

[10] V. Madisetti, D. B. Williams, eds., *The Digital Signal Processing Handbook,* Chapter 42, D. Sinha et al., "The Perceptual Audio Coder (PAC)," CRC Press, Boca Raton, Fl., 1998.

[11] ITU-R, "Methods for the subjective assessment of small impairments in audio systems including multichannel sound systems," Rec. ITU-R BS. 1116-1, Geneva, 1997

[12] U. Zölzer, *Digital Audio Signal Processing,* John Wiley & Sons, 1997.

[13] J. D. Johnston, "Estimation of perceptual entropy using noise masking criteria," in *Proc. ICASSP,* pp. 2524–2527, Apr. 1988.

[14] M. Bosi and R. E. Goldberg, *Introduction to Digital Audio Coding and Standards,* Kluwer Academic Publishers, 2002.

[15] E. Zwicker, H. Fastl, and H. Frater, *Psychoacoustics: Facts and Models,* Springer Verlag; 2nd edition, 1999.

[16] P. P. Vaidyanathan, *Multirate Systems and Filter Banks,* Prentice Hall, 1993.

[17] G. Schuller and T. Karp, "Modulated Filter Banks with Arbitrary system delay: efficient implementations and the time-varying Case," *IEEE Transactions on Signal Processing,* pp. 737–748, Mar. 2000.

[18] G. Schuller, "Time-varying filter banks with low delay for audio coding," 105th AES Convention, San Francisco, CA, Sept. 26-29, 1998.

[19] G. Schuller and M. J. T. Smith, "New framework for modulated perfect reconstruction filter banks,"*IEEE Transactions on Signal Processing,* vol. 44, pp. 1941 -1954, Aug. 1996.

[20] V. Madisetti and D. B. Williams (Editors) *The Digital Signal Processing Handbook,* by CRC Press, Book and CD-ROM edition, 1997.

[21] G. M. Phillips, "Echo and its effects on the telephone user," *Bell Laboratories Record,* vol. 32, pp. 281–284, Aug. 1954.

[22] G. Schuller and A. Harma, "Low delay audio compression using predictive coding," in *Proc. ICASSP,* Orlando, FL, May 13–17, 2002.

[23] J. Herre, "Temporal noise shaping, quantization and coding methods in perceptual audio coding: a tutorial introduction," in *AES 17th International Conference,* Florence, Italy, Sept. 2-5, 1999.

[24] E. Allamanche, R. Geiger, J. Herre, and T. Sporer, "MPEG-4 Low Delay Audio Coding based on the AAC Codec," 106th AES Convention, Munich, Germany, May, 1999.

[25] N. S. Jayant and P. Noll, *Digital Coding of Waveforms,* Prentice Hall, Englewood Cliffs, New Jersey, 1984.

[26] F. K. Soong and B.-H. Juang, "Line spectrum pair (LSP) and speech data compression," in *Proc. ICASSP,* 1984, pp. 1.10.1–1.10.4.

[27] G. Schuller, B. Yu, D. Huang, and B. Edler, "Perceptual audio coding using adaptive Pre- and post-filters and lossless compression," *IEEE Trans. Speech Audio Processing,* pp. 379–390, Sept. 2002.

[28] A. Gelman, H. Stein, and D. Rubin, *Bayesian Data Analysis,* New York : Chapman & Hall, 1995.

Chapter 12

SOUND FIELD SYNTHESIS

Sascha Spors
University of Erlangen–Nuremberg
spors@LNT.de

Heinz Teutsch
University of Erlangen–Nuremberg
teutsch@LNT.de

Achim Kuntz
University of Erlangen–Nuremberg
kuntz@LNT.de

Rudolf Rabenstein
University of Erlangen–Nuremberg
rabe@LNT.de

Abstract Conventional multichannel audio reproduction systems for entertainment or communication are not capable of immersing a large number of listeners in a well defined sound field. A novel technique for this purpose is the so-called wave field synthesis. It is based on the principles of wave physics and suitable for an implementation with current multichannel audio hard- and software components. A multiple number of fixed or moving sound sources from a real or virtual acoustic scene is reproduced in a listening area of arbitrary size. The listeners are not restricted in number, position, or activity and are not required to wear headphones. A successful implementation of wave field synthesis systems requires to address also spatial aliasing and the compensation of non-ideal properties of loudspeakers and of listening rooms.

Keywords: Wave Field Synthesis, Multichannel Audio, Loudspeaker Array, Loudspeaker Compensation, Room Compensation

1. INTRODUCTION

State-of-the-art systems for the reproduction of spatial audio suffer from a serious problem: The spatial properties of the reproduced sound can only be perceived correctly in a small part of the listening area, the so-called sweet spot. This restriction occurs because conventional reproduction of spatial audio is based on psychoacoustics, i.e. mainly intensity panning techniques. A solution to this problem calls for a new reproduction technique which allows the synthesis of physically correct wave fields of three-dimensional acoustic scenes. It should result in a large listening area which is not restricted to a particular sweet spot.

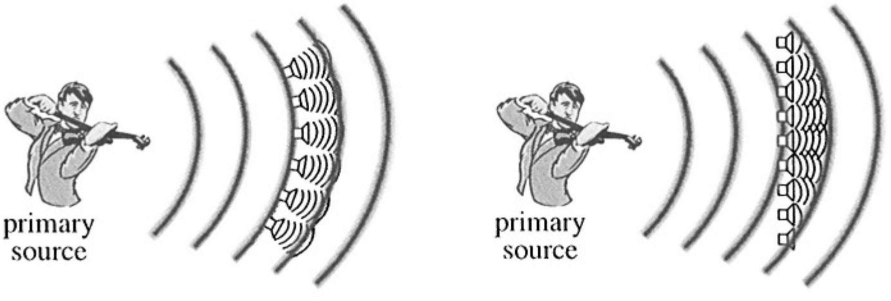
A number of different approaches have been suggested. They can be roughly categorized into advanced panning techniques, ambisonics, and wave field synthesis. Advanced panning techniques aim at enlarging the sweet spot well known from two-channel stereo or 5-channel surround sound systems. An example is the vector base amplitude panning technique (VBAP) as described e.g. in [1]. Ambisonics systems represent the sound field in an enclosure by an expansion into three-dimensional basis functions. A faithful reproduction of this sound field requires recording techniques for the contributions of all relevant basis functions. In common realizations, only the lowest order contributions are exploited [2].

This chapter presents the third approach from above, wave field synthesis (WFS). It is a technique for reproducing the acoustics of large recording rooms in smaller sized listening rooms. The spatial properties of the acoustical scene can be perceived correctly by an arbitrary large number of listeners which are allowed to move freely inside the listening area without the need of their positions being tracked. These features are achieved through a strict foundation on the basic laws of acoustical wave propagation.

WFS typically uses more loudspeakers than audio channels. Contrary to conventional spatial audio reproduction techniques like two-channel stereo or 5-channel surround, there exists no fixed mapping of the audio channels to the reproduction loudspeakers. Instead, a representation of the three-dimensional acoustic scene is used that can be reproduced by various WFS systems utilizing different loudspeaker setups.

The theory of WFS has been initially developed at the Technical University of Delft over the past decade [3, 4, 5, 6, 7] and has been further investigated in [8, 9, 10, 11, 12]. In [13] it has been shown how WFS is related to the Ambisonics approach.

Section 2 covers the physical foundations and the basic rendering technique. Two methods for sound field rendering are covered in detail in Section 3. The practical implementation requires a thorough analysis of acoustical wave fields which is shown in Section 4. It is the basis of methods for compensating the



(a) Synthesis of a wave front from spherical waves according to Huygens' principle.

(b) Synthesis of a wave front by a loudspeaker array with appropriately weighted and delayed driving signals.

Figure 12.1 **Basic principle of wave field synthesis.**

influence of non-ideal loudspeakers and of the listening room as described on Section 5. Finally the implementation of wave field synthesis systems is shown in Section 6.

2. RENDERING OF SOUND FIELDS WITH WAVE FIELD SYNTHESIS

This section gives an overview of the physical foundations of wave field synthesis and its practical realization.

2.1 PHYSICAL FOUNDATION OF WAVE FIELD SYNTHESIS

The theoretical basis of WFS is given by the Huygens' principle. Huygens stated that any point of a propagating wave at any instant conforms to the envelope of spherical waves emanating from every point on the wavefront at the prior instant. This principle can be used to synthesize acoustic wave fronts of arbitrary shape. Of course, it is not very practical to position the acoustic sources on the wave fronts for synthesis. By placing the loudspeakers on an arbitrary fixed curve and by weighting and delaying the driving signals, an acoustic wave front can be synthesized with a loudspeaker array. Figure 12.1 illustrates this principle.

The mathematical foundation of this more illustrative description of WFS is given by the Kirchhoff-Helmholtz integral (12.1), which can be derived by combining the acoustic wave equation and the Green's integral theorem [14]

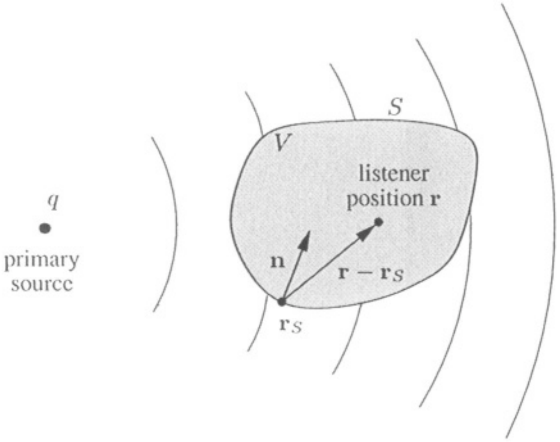


Figure 12.2 Parameters used for the Kirchhoff-Helmholtz integral (12.1).

$$P(\mathbf{r},\omega) = \frac{1}{4\pi}\oint_S \left[P(\mathbf{r}_S,\omega)\frac{\partial}{\partial\mathbf{n}}\left(\frac{e^{-j\beta|\mathbf{r}-\mathbf{r}_S|}}{|\mathbf{r}-\mathbf{r}_S|}\right) - \frac{\partial P(\mathbf{r}_S,\omega)}{\partial\mathbf{n}}\frac{e^{-j\beta|\mathbf{r}-\mathbf{r}_S|}}{|\mathbf{r}-\mathbf{r}_S|}\right] dS. \quad (12.1)$$

Figure 12.2 illustrates the parameters used. In (12.1), S denotes the surface of an enclosed volume V, $(\mathbf{r} - \mathbf{r}_S)$ the vector from a surface point $\mathbf{r}_S$ to an arbitrary listener position $\mathbf{r}$ within the source free volume V, $P(\mathbf{r}_S,\omega)$ the Fourier transform of the pressure distribution on S, and $\mathbf{n}$ the inwards pointing surface normal. The temporal angular frequency is denoted by ω, $\beta = \omega/c$ is the wave number, and c the speed of sound. The terms involving exponential functions describe monopole and dipole sources, i.e. the gradient term accompanying $P(\mathbf{r}_S,\omega)$ represents a dipole source distribution while the term accompanying the gradient of $P(\mathbf{r}_S,\omega)$ represents a monopole source distribution.

The Kirchhoff-Helmholtz integral states that at any listening point within the source-free volume V the sound pressure $P(\mathbf{r},\omega)$ can be calculated if both the sound pressure and its gradient are known on the surface enclosing the volume. Thus a wave field within the volume V can be synthesized by generating the appropriate pressuredistribution $P(\mathbf{r}_S,\omega)$ by dipole source distributions and its gradient by monopole source distributions on the surface S. This interrelation is used for WFS-based sound reproduction as discussed below.

2.2 WAVE FIELD SYNTHESIS BASED SOUND REPRODUCTION

Following the above presentation of the physical foundations, a first introduction to the techniques for WFS-based sound reproduction is given here.

The effect of monopole and dipole distributions described by (12.1) can be approximated by appropriately constructed loudspeakers. However, a direct realization of the scenario shown in Fig. 12.2 would require a very high number of two different types of loudspeakers densely spaced on the entire surface S. Since such an approach is impractical, several simplifications are necessary to arrive at a realizable system:

1. Degeneration of the surface S to a plane S_0 separating the primary sources and the listening area.

2. Reformulation of (12.1) in terms of only one kind of sources, e.g. either monopoles only or dipoles only.

3. Degeneration of the plane S_0 to a line.

4. Spatial discretization.

These simplifications are now discussed in more detail.

The first step maps the volume V to an entire half space such that the arbitrarily shaped surface S turns into a plane S_0. Then the sound field caused by the monopoles and dipoles is symmetric with respect to the plane S_0.

This symmetry is exploited in the second step. Since only the sound field on one side of S_0 is of interest, there is no need to control the sound fields on both sides independently of each other. Consequently, one kind of sources on S_0 can be eliminated. Dropping e.g. the dipoles and keeping the monopoles causes an even symmetry of the resulting sound field. Thus a considerable simplification results, since the desired sound field on one side of S_0 is generated by monopoles alone.

The results of steps 1 and 2 for monopoles are described by the Rayleigh I integral [6] as

$$P(\mathbf{r},\omega) = \rho c \frac{j\beta}{2\pi} \int_{S_0} \left[V_n(\mathbf{r}_S,\omega) \frac{e^{-j\beta|\mathbf{r}-\mathbf{r}_s|}}{|\mathbf{r}-\mathbf{r}_s|} \right] dS_0, \qquad (12.2)$$

where ρ denotes the static density of the air, c the speed of sound and V_n the particle velocity perpendicular to the surface. A similar description for dipoles only leads to the Rayleigh II integral [6].

The third step is motivated by the requirement to synthesize the wave field correctly only in the horizontal ear plane of the listener. Deviations from the true reconstruction well above and below the head of the listeners can be tolerated.

For this scenario the surface S_0 further degenerates to a line surrounding the listening area.

The fourth step is necessary because loudspeakers cannot be placed infinitely close to each other. Equidistant spatial discretization with space increment $\Delta\lambda$ gives the discrete monopole positions $\mathbf{r}_i$. Steps 3 and 4 result in a one-dimensional representation of the Rayleigh I integral (12.2) for discrete monopole positions [7], i.e.

$$P(\mathbf{r},\omega) = \sum_i \left[A_i(\omega) P(\mathbf{r}_i,\omega) \frac{e^{-j\beta|\mathbf{r}-\mathbf{r}_i|}}{|\mathbf{r}-\mathbf{r}_i|} \right] \Delta\lambda, \tag{12.3}$$

where $A_i(\omega)$ denotes a geometry dependent weighting factor.

Based on the above equation, an arbitrary sound field can be synthesized within a listening area leveled with the listeners ears. The monopole distribution on the line surrounding the listening area is approximated by an array of closed loudspeakers mounted at a distance $\Delta\lambda$. In the most simple case, several linear loudspeaker arrays are placed at appropriate angles to enclose the listening area fully or partly. Figure 12.7 shows a typical arrangement.

Up to now it has been assumed that no acoustic sources lie inside the volume *V*. The theory presented above can also be extended to the case where sources lie inside the volume *V* [5]. Then acoustical sources can be placed between the listener and the loudspeakers within the reproduction area (focused sources). This is not possible with traditional stereo or multichannel setups.

The simplifications described above lead to deviations from the true sound pressure distribution. Particularly, the effects of spatial discretization limit the performance of real WFS systems. In practice, two effects are observed:

1. Spatial aliasing.

2. Truncation effects.

Spatial aliasing is introduced through the discretization of the Rayleigh integral due to spatial sampling. The aliasing frequency is given by [7]

$$f_{\text{al}} = \frac{c}{2\Delta\lambda \sin\alpha_{\text{max}}}, \tag{12.4}$$

where α_{max} denotes the maximum angle of incidence of the synthesized plane wave relative to the loudspeaker array. Assuming a loudspeaker spacing of $\Delta\lambda = 10$ cm, the minimum spatial aliasing frequency is $f_{\text{al}} = 1700$ Hz. Regarding the standard audio bandwidth of 20 kHz spatial aliasing seems to be a problem for practical WFS systems. Fortunately, the human auditory system seems not to be very sensitive to these aliasing artifacts.

Truncation effects appear when the line is represented by a loudspeaker array of finite extension. They can be understood as diffraction waves propagating from the ends of the loudspeaker array. Truncation effects can be minimized, e.g. by filtering in the spatial domain (tapering, windowing) [7].

The remaining sections of this chapter show how to implement the physical processes described by the Rayleigh I integral by discrete-time, discrete-space algorithms. For the sake of readability of the text, some simplifications of the notation are adopted. Most relations are given in terms of Fourier transformations with respect to the continuous time t. The corresponding frequency variable in the Fourier domain is denoted by ω. Discrete-time operations (e.g. convolutions) are given in terms of the discrete-time variable k. Discrete-time functions (sequences) are distinguished from their continuous-time counterparts by brackets (e.g. $h[k]$). For simplicity, no corresponding discrete-time Fourier transformation is introduced. Instead, the Fourier transformation is used. It is assumed that the reader is familiar with the basic continuous-to-discrete and time-to-frequency transitions.

3. MODEL-BASED AND DATA-BASED RENDERING

The Rayleigh I integral (12.3) formulates a mathematical description of the wave fields that are synthesized by linear loudspeaker arrays. The loudspeaker driving signals can be derived by two different specializations of the Rayleigh I integral which result in two rendering techniques suited for different applications of WFS systems. These two rendering techniques are described in the next sections.

3.1 DATA-BASED RENDERING

In (12.3) the expression $e^{-j\beta|\mathbf{r}-\mathbf{r}_i|}/|\mathbf{r}-\mathbf{r}_i|$ describes the sound pressure propagating from the ith loudspeaker to the listener position $\mathbf{r}$. Therefore, the loudspeaker driving signals $W_i(\omega)$ for arbitrary wave fields can be computed according to (12.3) as

$$W_i(\omega) = A_i(\omega)P(\mathbf{r}_i, \omega). \tag{12.5}$$

The pressure distribution $P(\mathbf{r}_i, \omega)$ contains the entire information of the sound field produced at the loudspeaker position $\mathbf{r}_i$ by an arbitrary source q (see Fig. 12.2). The propagation from the source to the loudspeaker position $\mathbf{r}_i$ can be modeled by a transfer function $H_i(\omega)$. By incorporating the weighting factors $A_i(\omega)$ into $H_i(\omega)$ we can calculate the loudspeaker driving signals as

$$W_i(\omega) = H_i(\omega)Q(\omega). \tag{12.6}$$

$Q(\omega)$ denotes the Fourier transform of the source signal q. Transforming this equation back into the discrete-time domain, the vector of loudspeaker driving signals $\mathbf{w}[k] = [w_1, \ldots, w_i, \ldots, w_M]^{\mathrm{T}}$ can be derived by a multichannel con-

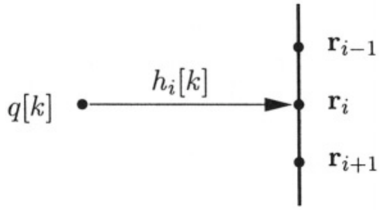


Figure 12.3 Principle of data-based rendering. $h_i[k]$ denotes an appropriate impulse response that is convolved with the source signal $q[k]$ to derive the ith loudspeaker driving signal.

volution of measured or synthesized impulse responses with the source signal $q[k]$. This reproduction technique is illustrated in Fig. 12.3.

Arranging the discrete-time transforms of $H_i(\omega)$, i.e. the impulse responses $h_i[k]$, in a vector $\mathbf{h}[k] = [h_1, \ldots, h_i, \ldots, h_M]^{\mathrm{T}}$, the loudspeaker driving signals can be expressed as

$$\mathbf{w}[k] = \mathbf{h}[k] * q[k], \tag{12.7}$$

where the symbol $*$ denotes the time-domain convolution operator.

The impulse responses for auralization cannot be obtained the conventional way by simply measuring the response from a source to a loudspeaker position. In addition to the sound pressure also the particle velocity is required to extract the directional information. This information is necessary to take into account the direction of the traveling waves for auralization. The room impulse responses have to be recorded by special microphone setups and extrapolated to the loudspeaker positions as shown in Sec. 4.

Data-based rendering allows for a very realistic reproduction of the acoustics of the recording room. The main drawback of this reproduction technique is the high computational load caused by the multichannel convolution (12.7), especially when using long room impulse responses. Data-based rendering is typically used for high-quality reproduction of static scenes.

3.2 MODEL-BASED RENDERING

Model-based rendering may be interpreted as a special case of the data-based approach: Instead of measuring room impulse responses, models for acoustic sources are used to calculate the loudspeaker driving signals. Point sources and plane waves are the most common models used in model-based WFS systems.

Based on these models, the transfer functions $H_i(\omega)$ in (12.6) can be obtained in closed form. For a point source, $H_i(\omega)$ can be derived as [7]

$$H_i(\omega) = K\sqrt{\frac{j\beta}{2\pi}}\,\frac{e^{-j\beta|\mathbf{r}_i - \mathbf{r}_q|}}{\sqrt{|\mathbf{r}_i - \mathbf{r}_q|^3}}\Delta\lambda. \tag{12.8}$$

K is a geometry dependent constant and $\mathbf{r}_q$ denotes the position of the point source. Transforming $W_i(\omega)$ back into the time domain and employing time discretization, the loudspeaker driving signals can be computed from the source signal $q[k]$ by delaying, weighting and filtering, i.e.

$$w_i[k] = c_i \; (\, g[k] * \delta[k-\kappa] \,) * q[k], \tag{12.9}$$

where c_i and κ denote an appropriate weighting factor and delay respectively, and $g[k]$ is the inverse Fourier transform of $\sqrt{j\beta/2\pi}$. Multiple (point) sources can be synthesized by superimposing the loudspeaker signals for each source.

Similar models exist for plane wave sources. Plane waves and point sources can be used to simulate classical loudspeaker setups, like stereo and 5-channel systems. Thus, WFS is backward compatible to existing sound reproduction systems. It can even improve them by optimal setup of the virtual loudspeakers in small listening rooms.

Model-based rendering results in lower computational load compared to data-based rendering because convolutions with long room impulse responses are not required. Rendering of moving sound sources does not call for significant additional effort and also reproduction of virtual sources in front of the loudspeaker array (i. e. inside the listening area) is possible. The characteristics of the recording room can be simulated by placing additional point sources that model acoustic reflections (e. g. image method [15]). The model-based approach is especially interesting for synthetic or parameterized acoustic scenes.

3.3 HYBRID APPROACH

For scenes with long reverberation time, data-based WFS systems require a lot of computational resources. In such applications a hybrid approach can be used that achieves similar sound quality with reduced computational complexity.

Here, the direct sound is reproduced by a virtual point source located at the desired source position. The reverberation is reproduced by synthesizing several plane waves from various directions to obtain a perceptually diffuse reverberant sound field. The computationally demanding convolution has to be performed only for each plane wave direction instead of each individual loudspeaker as would be necessary for data-based WFS systems. Some realizations of WFS systems successfully synthesize the reverberation with about eight plane waves [8, 16].

4. WAVE FIELD ANALYSIS

Data-based rendering as introduced in the previous section requires to gain knowledge about the impulse responses $\mathbf{h}[k]$ needed for auralization. In principle it is possible to measure the impulse responses by placing microphones at the loudspeaker positions in the scene to be recorded and measuring the impulse responses from the source(s) to the microphones [14]. This approach has one major drawback: The recorded scene dataset is limited to one particular loudspeaker setup. A higher degree of flexibility could be obtained by analyzing the acoustic wave field inside the entire area of interest, which could then be reproduced by arbitrary loudspeaker setups. Wave field analysis techniques are used to analyze the acoustic wave field for a region of interest. The following section introduces the wave field analysis techniques used in the context of WFS. These techniques are mainly derived from seismic wave theory [17].

In general, there will be no access to the entire two-dimensional pressure field $P(\mathbf{r},\omega)$ for analysis. However, the Kirchhoff-Helmholtz integral (12.1) allows to calculate the acoustic pressure field $P(\mathbf{r},\omega)$ from the sound pressure and its gradient on the surface enclosing the desired field and vice versa (see Section 2.1). Therefore, the Kirchhoff-Helmholtz integral (12.1) can not only be used to introduce the concept of wave field synthesis, but also to derive an efficient and practical way to analyze wave fields. We will introduce the analysis tools for two-dimensional wave fields in the following. This requires a two-dimensional formulation of the Kirchhoff-Helmholtz integral, as given e.g. in [17]. As a result, it is sufficient to measure the acoustic pressure and its gradient on a line surrounding the area of interest in order to capture the acoustics of the entire area.

To proceed with the analysis of wave fields, a closer look at the acoustic wave equation is required. The eigensolutions of the acoustic wave equation in three dimensional space appear in different forms, depending on the type of the adopted coordinate system. In a spherical coordinate system the simplest solution of the wave equation are spherical waves, while plane waves are a simple solution in Cartesian coordinates. Of course, plane waves can be expressed as a superposition of spherical waves (see Huygens' principle) and vice versa [13]. Accordingly, two major types of transformations exist for three dimensional wave fields, the decomposition of the field into spherical harmonics or into plane waves. The decomposition into spherical harmonics is used in the Ambisonics system [2] whereas the decomposition into plane waves is often utilized for WFS. We will introduce the decomposition into plane waves in the following. To simplify matters in the discussion it is assumed that the region of interest is source free. See [18] for a more general view.

The next step is to transform the pressure field $p(\mathbf{r},t)$ into plane waves with the incident angle θ and the temporal offset τ with respect to an arbitrary reference point. This way any recorded data $p(\mathbf{r},t)$ become independent from the geometry used for recording. This transformation is called *plane wave decomposition* and is given by

$$\tilde{p}(\theta,\tau) = \mathcal{P}\left\{p(\mathbf{r},t)\right\}, \tag{12.10}$$

where $\tilde{p}(\theta, \tau)$ denotes the plane wave decomposed wave field and $\mathcal{P}$ the plane wave decomposition operator. The plane wave decomposition maps the pressure field into an angle, offset domain. This transformation is also well known as the *Radon transformation* [19] from image processing. The Radon transformation maps straight lines in the image domain into Dirac peaks in the Radon domain. It is therefore typically used for edge detection in digital image processing. The same principle applies for acoustic fields: A plane wave can be understood as an 'edge' in the pressure field $p(\mathbf{r}, t)$.

One of the benefits using a spatial transform of the wave field is, that plane waves can be extrapolated easily to other positions [17]

$$P(r,\theta,\omega) = \mathcal{P}^{-1}\left\{\tilde{P}(\theta,\omega)\right\} = \int_0^{2\pi} \tilde{P}(\theta',\omega)e^{-j\beta r\cos(\theta-\theta')}d\theta', \quad (12.11)$$

where r and θ denote the position in cylindrical coordinates with respect to the origin of the plane wave decomposition. This allows, in principle, to extrapolate a recorded wave field to arbitrary points without loss of information. Wave field extrapolation can be used to extrapolate a measured field to the loudspeaker positions for reproduction purposes or to create a complete image of the captured sound field. The impulse responses required for data-based rendering (see Section 3.1) can be obtained by measuring the plane wave decomposed impulse response of the wave field and extrapolation to the loudspeaker positions.

The main benefit of the plane wave decomposition is that it is possible to capture the acoustical characteristics of a whole area through a measurement on the boundary of this area. Plane wave decomposed impulse responses are therefore an efficient tool to describe the acoustical properties of a whole area of interest.

For practical reasons the acoustic pressure and its gradient will only be measured on a limited number of positions on the circle. This spatial sampling can result in spatial aliasing in the plane wave decomposed field if the sampling theorem is not taken into account. An efficient implementation of the plane wave decomposition for circular microphone arrays can be found in [18]. Pressure and pressure gradient microphones placed on a circle surrounding the area of interest are used, together with the duality between plane waves and cylindrical harmonics for this realization.

5. LOUDSPEAKER AND LISTENING ROOM COMPENSATION

The theory of WFS systems as described in Section 2. was derived assuming free field propagation of the sound emitted by the loudspeakers. However, real listening rooms do not exhibit free field propagation nor do real loudspeakers act like point sources. Therefore, the compensation of the non-ideal effects of listening rooms and loudspeakers is required to arrive at practical WFS systems.

Acoustic reflections off the walls of the listening room can degrade the sound quality, especially the perceptibility of the spatial properties of the auralized acoustic scene. Listening room compensation aims at improving the perceived quality in sound reproduction in non-anechoic environments.

Additionally, the theory of WFS assumes that the secondary sources act like ideal monopole sources. Real loudspeakers however, behave more or less different from this idealistic assumption. They show deviations from the monopole characteristic as well as deviations from an ideally flat frequency response. This holds especially for the loudspeakers used for WFS systems because small dimensions are mandatory in order to arrive at a reasonable aliasing frequency (see Section 2.1). Loudspeaker compensation aims at reducing the effects caused by the non-ideal characteristics of the loudspeakers used for reproduction. The next sections introduce possible solutions to room and loudspeaker compensation.

5.1 LISTENING ROOM COMPENSATION

When listening to a recording that contains the reverberation of the recorded scene, the reverberation caused by the listening room interferes with the recorded sound field in a potentially negative way [20]. Figure 12.4 shows the effect of the listening room on the auralized wave field. The WFS system in the listening room composes the recorded acoustic scene by positioning the dry source signals at their respective positions relative to the recording room together with the acoustic effect of the recording room. The size and position of the recording room relative to the listening room is shown by a dashed line for reference. The dashed lines from one virtual source to one exemplary listening position show the acoustic rays for the direct sound and one reflection off the side wall of the virtual recording room. The solid line from one loudspeaker to the listening position shows a possible reflection of the loudspeaker signal off the wall of the listening room. This simple example illustrates the effect of the listening room on the auralized wave field. Perfect listening room compensation would entirely eliminate the effects caused by the listening room.

Room compensation can be realized by calculating a set of suitable compensation filters to pre-filter the loudspeaker signals. Unfortunately there are a number of pitfalls when designing a room compensation system. A good overview of classical room compensation approaches and their limitations can be found in [21]. Most of the classical approaches fail to provide room compensation for a large listening area [22, 23]. There are three main reasons for this failure: problems involved with the calculation of the compensation filters, lack of directional information for the wavefronts and lack of control over the wave field inside the listening area. Most classical room compensation systems measure the impulse responses from one or more loudspeakers to one or more

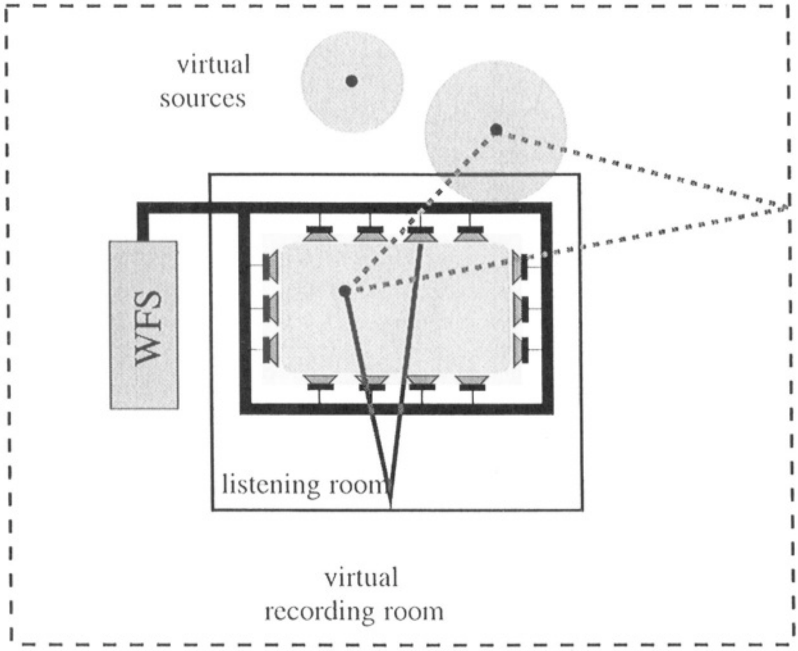


Figure 12.4 Simplified example that shows the effect of the listening room on the auralized wave field. The dashed lines from one virtual source to one exemplary listening position show the acoustic rays for the direct sound and one reflection off the side wall of the virtual recording room. The solid line from one loudspeaker to the listening position shows a possible reflection of the loudspeaker signal off the wall of the listening room.

microphone positions. Unfortunately typical room impulse responses are, in general, non-minimum phase [24] which prohibits to calculate exact inverses for each measured impulse response. The measurements often involve only the acoustic pressure at the point they where captured. Without directional information on the traveling wave fronts a room compensation system will fail to determine which loudspeakers to use for compensation. Additionally most classical systems do not provide sufficient control over the wave field. They utilize only a limited number of loudspeakers. WFS may provide a solution for the last two problems concerning the lack of directional information for the traveling wave fronts and lack of control over the wave field.

5.1.1 Room Compensation for Wave Field Synthesis. As wave field synthesis in principle allows to control the wave field within the listening area it can also be used to compensate for the reflections caused by the listening room. Of course, this is only valid up to the spatial aliasing frequency (12.4) of the particular WFS system used. Above the aliasing frequency there is no control over the wave field possible. Destructive interference to compensate

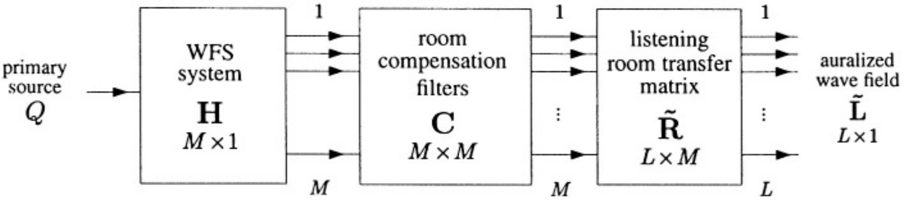


Figure 12.5 Block diagram of a WFS system including the influence of the listening room and the compensation filters to cope with the listening room reflections

for the listening room reflections will fail here. Some indications for what can be done above the aliasing frequency will be given in Section 5.1.2.

In the following a possible solution to room compensation with WFS will be introduced, more details can be found in [9, 11, 12]. Figure 12.5 shows the signal flow diagram of a room compensation system utilizing WFS. The primary source signal $q[k]$ is first fed through the WFS system operator $\mathbf{h}[k]$ (see Section 3). The influence of the listening room is compensated by compensation filters $\mathbf{C}[k]$. The resulting signals are then played back through the M loudspeakers of the WFS system. The loudspeaker and listening room characteristics are contained in the matrix $\tilde{\mathbf{R}}[k]$. Instead of using the microphone signals directly we perform a plane wave decomposition of the measured wave field into L plane waves as described in Section 4. The transformed wave field is denoted as $\tilde{\mathbf{R}}$. Using the plane wave decomposed wave fields instead of the microphone signals has the advantage that the complete spatial information about the listening room influence is included. This allows to calculate compensation filters which are in principle valid for the complete area inside the loudspeaker array, except for secondary effects not covered here that limit the performance [25].

The auralized wave field inside the listening area is expressed by the vector $\tilde{\mathbf{l}}[k]$. According to Fig. 12.5, the auralized wave field $\tilde{\mathbf{L}}(\omega)$ is given as

$$\tilde{\mathbf{L}}(\omega) = \tilde{\mathbf{R}}(\omega) \cdot \mathbf{C}(\omega) \cdot \mathbf{H}(\omega) \cdot Q(\omega). \tag{12.12}$$

In principle, perfect listening room compensation would be obtained if

$$\tilde{\mathbf{R}}(\omega) \cdot \mathbf{C}(\omega) = \tilde{\mathbf{F}}(\omega), \tag{12.13}$$

where $\tilde{\mathbf{F}}$ denotes the plane wave decomposed wave field of each loudspeaker under free field conditions.

Using the multiple-input/output inversion theorem (MINT) it is possible to solve the above equation exactly under certain realistic conditions as shown in [26]. However, many applications only require to compensate for the first reflections of the room. Using a least-squares error (LSE) method to calculate the compensation filters results in approximated inverse filters with shorter length compared to the MINT.

An efficient implementation of room compensation can be derived by integrating the WFS operator into the room compensation filters. Off-line calculation of $\mathbf{C}(\omega) \cdot \mathbf{H}(\omega)$ results in only M filters that have to be applied in real-time to the source signal. This is significantly less than in the approach presented above.

5.1.2 Compensation Above the Aliasing Frequency. Above the aliasing frequency of the WFS system no destructive interference can be used to compensate for the reflections caused by the listening room. Perfect compensation in the sense of a resulting free field wave propagation for each loudspeaker is therefore not possible above the aliasing frequency. A possible approach includes the use of psychoacoustic properties of the human auditory system to mask listening room reflections as indicated in [9].

5.2 LOUDSPEAKER COMPENSATION

Typically, two different types of loudspeakers are used to build arrays for WFS. The first type are classical cone loudspeakers and the second one are multiexciter panels.

Cone loudspeakers as used for WFS suffer from the small dimensions required for a reasonable aliasing frequency of the WFS system. Broadband cone loudspeakers normally show a narrowed radiation pattern on the main axis at high frequencies [5] and a non-ideal frequency response. Most of these problems can be solved to some extend by using high-quality (two-way) loudspeakers.

For the second type of loudspeakers used for WFS systems, multiexciter panels (MEP) [27], these non-ideal effects are much more severe. MEPs have a relatively simple mechanical construction. They consist of a foam board with both sides covered by a thin plastic layer. Multiple electro-mechanical exciters are glued equally spaced on the backside of the board. The board is fixed on the sides and placed in a damped housing. Because of this simple mechanical construction their frequency response is quite disturbed by resonances and notches. The benefits of MEPs are the simple construction and the possibility of seamless integration into walls.

Figure 12.6 shows the magnitude of the measured on-axis frequency responses of four exciter positions on a MEP prototype. As can be seen from the measured frequency responses, they do not only differ significantly from an ideally flat frequency response, they also show a position dependency. Multi-

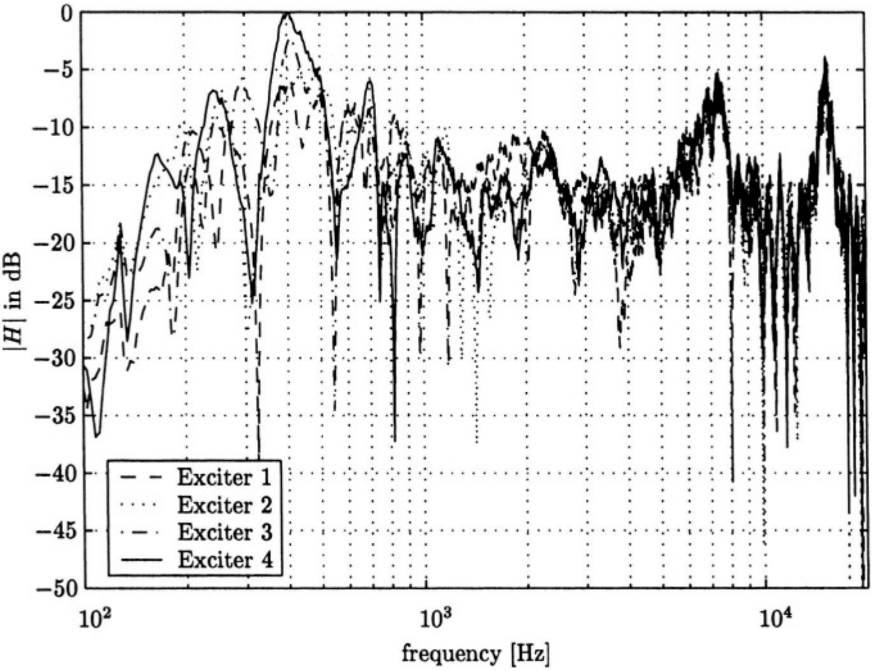


Figure 12.6 **On-axis magnitude frequency responses of four exciter positions on a multiexciter panel (MEP).**

channel loudspeaker compensation is therefore mandatory when using MEPs for auralization purposes.

Loudspeaker compensation can be seen as a subset of room compensation in our framework. By measuring the ’room’ impulse responses in an anechoic chamber the loudspeaker characteristics can be captured and compensated the same way as described in Section 5.1. The compensation filters include the compensation of the radiation pattern using the neighboring loudspeakers as well as the compensation of the frequency response and reflections caused by the panels itself.

However, this direct approach has one major drawback: The inverse filters include the compensation of potentially deep dips in the frequency response with high energy peaks for some frequencies. This can lead to clipping of the amplifiers or overload of the electro-mechanical system of the exciters. This problem can be solved by performing a smoothing of the magnitude of the recorded frequency responses. Additionally, the length of the inverse filters can be reduced by splitting the impulse responses into their non-minimum phase and minimum phase parts and using only the minimum phase part for the inversion process.

As for listening room compensation these approaches are only valid up to the aliasing frequency of the WFS system. Above the aliasing frequency there is no control over the produced wave field possible. Only individual compensation of the frequency response of each exciter can be performed here. A complete algorithm using the above described techniques is presented, e.g., in [10].

6. DESCRIPTION OF A SOUND FIELD TRANSMISSION SYSTEM

Figure 12.7 shows a typical WFS-based sound field recording and reproduction system. The recording room contains equipment for dry sound recording of the primary sound sources, their positions, and room impulse responses. As stated earlier, these impulse responses, in general, cannot be obtained by applying conventional room impulse response measurement procedures but need to be retrieved by using the techniques as described in Section 4.

Returning to Fig. 12.7, the size and position of the listening room relative to the recording room is shown by dashed lines. The loudspeaker array in the listening room is depicted schematically as a closed line surrounding the listening area (highlighted in gray). Note that, in this context, any setup of linear loudspeaker arrays can be utilized for sound field reproduction, as long as the inter-loudspeaker spacing remains constant.

The WFS system in the listening room composes an acoustic scene by positioning the dry source signals at their respective positions relative to the recording room and by taking into account the acoustic properties of the room to be auralized. The virtual sources created by this process may lie outside the physical dimensions of the listening room. This creates the impression of an enlarged acoustic space.

The following sections briefly detail typical realizations for the recording and reproduction of sound fields.

6.1 ACQUISITION OF SOURCE SIGNALS

Up to this point, one link in the chain that leads to realistic sound stage reproduction using wave field synthesis has been silently ignored, namely the acquisition of the source signals.

Flexible WFS-based rendering requires each source to be recorded in a way such that the influence of the room acoustics and other possibly competing sound sources, i.e. crosstalk between the acquired audio channels, is minimized. This concept is required for taking advantage of one of the central properties of WFS-based rendering, which is the ability to realistically auralize an acoustic scene in a different acoustic environment than it was originally recorded in.

For traditional multichannel recordings, there exist a vast variety of well established main- and multi-microphone techniques that are being utilized by

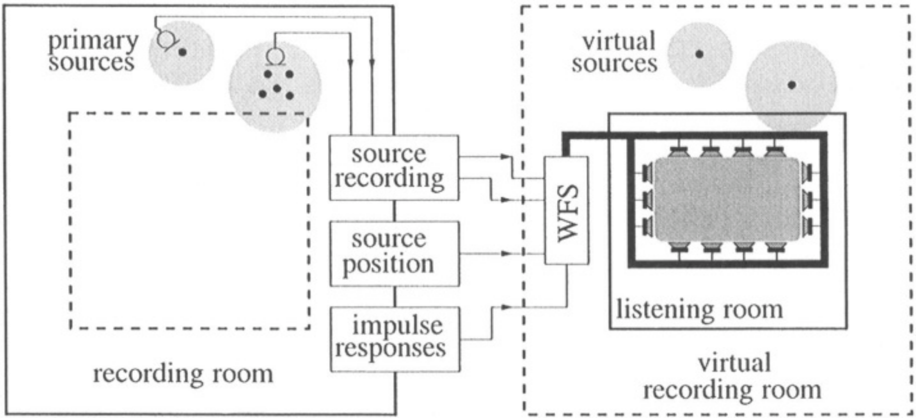


Figure 12.7 A WFS-based sound field recording and reproduction system

Tonmeisters for high-quality sound acquisition, e.g. [28]. In the WFS context, however, one of the main disadvantage of many traditional recording techniques is the natural constraint on spatial resolution which may limit WFS performance.

To combat this drawback, microphone array technology and associated signal processing can be used for obtaining enhanced spatial selectivity. Especially during live performances, microphone arrays have the further potential advantage of relieving the actors/musicians of having to carry or wear close-up microphones. Microphone arrays also may be able to add a new level of flexibility and creativity to the post-production stage by means of digital beamforming.

Ideally, for high-quality WFS-based applications, a general-purpose microphone array-based sound recording system should meet several requirements including

- high and flexible spatial selectivity,
- very low distortion and self noise,
- immunity against undesired ambient noise and reverberation,
- low-delay and real-time operation,
- frequency-independent beampattern, and
- sonic quality comparable to studio microphones.

Unfortunately, the third item describes a yet unsolved problem. However, promising microphone array designs have been proposed that meet many of the other requirements (see e.g. Chap. 3 in this book, or [29]), and can therefore be

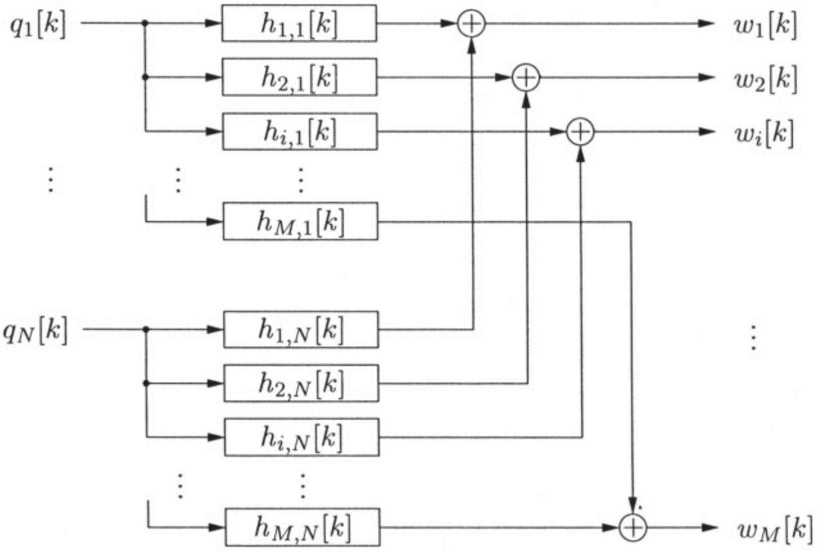


Figure 12.8 Block diagram of WFS system for reproduction of N sources using M loudspeakers. Depending on the application M may range from several tens to several hundred loudspeakers.

utilized as a recording front-end for WFS-based systems in designated recording environments.

As indicated in Fig. 12.7, WFS-based reproduction systems need to acquire information about the source positions for spatially correct and realistic sound stage reproduction. Again, microphone array systems implementing advanced acoustic source localization algorithms can be of benefit for this task. The interested reader is referred to Chapters 8 and 9 in this book.

6.2 SOUND STAGE REPRODUCTION USING WAVE FIELD SYNTHESIS

In an actual implementation of a WFS reproduction system, the loudspeaker driving signals of the array, $\mathbf{w}[k]$, are synthesized from the source signals $\mathbf{q}[k] = [q_1, \ldots, q_n, \ldots, q_N]^{\mathrm{T}}$ by convolution operations with a set of filters $h_{i,n}[k]$ as shown in Fig. 12.8. Note that this figure depicts a multi-source scenario in contrast to the theory presented in the previous sections where a single source has been assumed for auralization. The multi-source scenario can be derived from the single-source case by simply applying the superposition principle.

Depending on the application, the filters not only include source positions and the acoustics of the recording room, which have been acquired during the recording stage (see Fig 12.7), but also loudspeaker and listening room compensation filters. In the simplest case of model-based rendering the filter coefficients, $h_{i,n}[k]$, can be directly derived from (12.9).

A WFS-based reproduction system needs to process and reproduce, often in real-time, a high number of audio channels simultaneously. Often, the focus is on a PC-based implementation in contrast to DSP-based (embedded) solutions, because PCs potentially preserve a high degree of freedom for development platforms.

On a hardware level, a typical WFS system consists of one or more PCs performing all the required digital signal processing, multichannel digital audio interfaces, D/A-converters, amplifiers, and loudspeaker systems. State-of-the-art PCs nowadays allow for real-time digital signal processing of a high number of audio channels. Standard operating systems and high-level programming languages enable rapid implementation of the algorithms as well as graphical user interfaces. All the necessary hardware components are, in principle, commercially available at the time this book was printed. The design of special purpose hardware for WFS systems has the benefit of higher integration, more specialized functionality and lower cost. An example could be the design of multichannel power amplifiers with integrated D/A converters and digital signal processors.

7. SUMMARY

The wave field synthesis technology described in this chapter is capable of reproducing acoustic scenes for communication and entertainment applications. It is based on the basic laws of acoustical wave propagation. The required transition from the physical description to a discrete-time, discrete-space realization with current multichannel audio technology is well understood. The main components of this transition can be described in terms of geometrical considerations, spatial transformations, temporal and spatial sampling, and multichannel equalization.

The basic signal processing structure in the form of a multiple-input, multiple-output convolution allows a unified system implementation, even if different kinds of rendering approaches are used. This unified structure for the implementation on the one hand and the flexibility and adaptability to different applications on the other hand makes wave field synthesis a suitable technology for sound field rendering.

References

[1] V. Pulkki, "Compensating displacement of amplitude-panned virtual sources," in *Proc. of the AES 22nd Int. Conference,* Audio Engineering Society, 2002, pp. 186–195.

[2] M. A. Gerzon, "Ambisonics in multichannel broadcasting and video," *J. Acoust. Soc. Am.,* vol. 33, pp. 859–871, Nov. 1985.

[3] A. J. Berkhout, "A holographic approach to acoustic control," *J. of the Audio Engineering Society,* vol. 36, pp. 977–995, Dec. 1988.

[4] E. W. Start, *Direct Sound Enhancement by Wave Field Synthesis.* PhD thesis, Delft University of Technology, 1997.

[5] E. N. G. Verheijen, *Sound Reproduction by Wave Field Synthesis.* PhD thesis, Delft University of Technology, 1997.

[6] P. Vogel, *Application of Wave Field Synthesis in Room Acoustics.* PhD thesis, Delft University of Technology, 1993.

[7] D. de Vries, E. W. Start, and V.G. Valstar, "The wave field synthesis concept applied to sound reinforcement: restrictions and solutions," in *96th AES Convention,* Audio Engineering Society (AES), Amsterdam, Netherlands, Feb. 1994.

[8] E. Hulsebos and D. de Vries, "Parameterization and reproduction of concert hall acoustics with a circular microphone array," in *112th AES Convention,* Audio Engineering Society, Munich, Germany, May 2002.

[9] E. Corteel and R. Nicol, "Listening room compensation for wave field synthesis. What can be done?" in *23rd AES International Conference,* Audio Engineering Society, Copenhagen, Denmark, May 2003.

[10] E. Corteel, U. Horbach, and R. S. Pellegrini, "Multichannel inverse filtering of multiexciter distributed mode loudspeakers for wave field synthesis," in *112th AES Convention,* Audio Engineering Society, Munich, Germany, May 2002.

[11] S. Spors, A. Kuntz, and R. Rabenstein, "An approach to listening room compensation with wave field synthesis," in *AES 24th International Conference on Multichannel Audio,* Audio Engineering Society, Banff, Canada, June 2003, pp. 49–52.

[12] S. Spors, A. Kuntz, and R. Rabenstein, "Listening room compensation for wave field synthesis," in *IEEE International Conference on Multimedia and Expo (ICME),* Baltimore, USA, July 2003, pp. 725–728.

[13] R. Nicol and M. Emerit, "Reproducing 3D-sound for videoconferencing: A comparison between holophony and ambisonic," in *First COST-G6 Workshop on Digital Audio Effects (DAFX98)*, Barcelona, Spain, Nov. 1998.

[14] A. J. Berkhout, D. de Vries, and P. Vogel, "Acoustic control by wave field synthesis," *J. Acoust. Soc. Am.,* vol. 93, pp. 2764–2778, May 1993.

[15] J. B. Allen and D.A. Berkley, "Image method for efficiently simulating small-room acoustics," *J. Acoust. Soc. Am.,* vol. 65, pp. 943–950, 1979.

[16] J.-J. Sonke and D. de Vries, "Generation of diffuse reverberation by plane wave synthesis," in *102nd AES Convention,* Audio Engineering Society, Munich, Germany, Mar. 1997.

[17] A. J. Berkhout, *Applied Seismic Wave Theory.* Elsevier, 1987.

[18] E. Hulsebos, D. de Vries, and E. Bourdillat, "Improved microphone array configurations for auralization of sound fields by wave field synthesis," in *110th AES Convention,* Audio Engineering Society, Amsterdam, Netherlands, May 2001.

[19] S. R. Deans, *The Radon Transform and Some of its Applications,* Krieger Publishing Company, 1993.

[20] E. J. Völker, W. Teuber, and A. Bob, "5.1 in the living room - on acoustics of multichannel reproduction," in *Proc. of the Tonmeistertagung,* Hannover, Germany, 2002.

[21] L. D. Fielder, "Practical limits for room equalization," in *111th AES Convention,* Audio Engineering Society, New York, NY, USA, Sept. 2001.

[22] J. N. Mourjopoulos, “Digital equalization of room acoustics,” *J. Audio Eng. Soc.,* vol. 42, pp. 884–900, Nov. 1994.

[23] F. Talantzis and D. B. Ward, “Multi-channel equalization in an acoustic reverberant environment: establishment of robustness measures,” in *Institute of Acoustics Spring Conference,* Salford, UK, Mar. 2002.

[24] S. T. Neely and J. B. Allen, “Invertibility of a room impulse response,” *J. Acoust. Soc. Am.,* vol. 66, pp. 165–169, July 1979.

[25] J.-J. Sonke, D. de Vries, and J. Labeeuw, “Variable acoustics by wave field synthesis: a closer look at amplitude effects,” in *104th AES Convention,* Audio Engineering Society, Amsterdam, Netherlands, May 1998.

[26] M. Miyoshi and Y. Kaneda, “Inverse filtering of room acoustics,” *IEEE Trans, on Acoustics, Speech, and Signal Processing,* vol. 36, pp. 145–152, Feb. 1988.

[27] M. M. Boone and W. P. J. Bruijn, “On the applicability of distributed mode loudspeakers panels for wave field synthesis based sound reproduction,” in *108th AES Convention,* Audio Engineering Society, Paris, France, Feb. 2000.

[28] W. Dooley and R. Streicher, “Basic stereo microphone perspectives - a review,” *J. of the AES*, vol. 33, July/Aug. 1985.

[29] H. Teutsch, S. Spors, W. Herbordt, W. Kellermann, and R. Rabenstein, “An integrated real-time system for immersive audio applications,” in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, USA, Oct. 2003.

Chapter 13

VIRTUAL SPATIAL SOUND

Carlos Avendano
Creative Advanced Technology Center
carlos@atc.creative.com

Abstract Future tele-collaboration systems need to support seamless spatial sound reproduction to achieve realistic immersion. In this chapter, we present techniques based on binaural signal processing, capable of encoding and rendering sound sources accurately in three-dimensional space using only two recording/playback channels, thus creating the illusion of a *virtual* source in space. The technique exploits the mechanisms that help to decode spatial information in the auditory system. An overview of the encoding and decoding mechanisms is given and their practical application to the design of virtual spatial sound (VSS) systems is discussed.

Keywords: Binaural Hearing, Binaural Technology, Virtual Sound, Spatial Sound, HRTF, HRIR, Crosstalk Cancellation

1. INTRODUCTION

In this chapter, we are concerned with the rendering of sound sources in three-dimensional space using only two playback channels. We focus our study on a family of techniques based on the synthesis of binaural signals. Given that the human hearing mechanism is binaural, all spatial information available to the listener is encoded within two acoustic signals (i.e. one to each ear). Thus in principle, only two playback channels are necessary and sufficient to realistically render sound sources localized at any point in space. Since the sound is not perceived as coming from any given loudspeaker the technique creates the illusion of a *virtual* sound source in free space.

There are many ways to create the impression that a sound source is located at a given point in space away from the loudspeakers. One familiar example is the playback over two speakers commonly known as *stereo* reproduction, where a sound source can be positioned within the angle subtended by the loudspeakers by simply manipulating the gains assigned to the source in each channel (i.e.

amplitude panning). If the desired rendering position of a source is outside of this region, then the angle between speakers needs to be increased or more rendering channels (and corresponding gain reassignments) have to be added. The success of amplitude panning depends highly on the location of the listener with respect to the loudspeakers. The location where amplitude panning (or any other playback technique) is most effective is commonly known as the *sweet spot.* Another technique, which renders virtual spatial sound accurately and has a large sweet spot, but which requires multiple playback channels is wavefield synthesis. This technique is described in Chapter 12.

The advantage of binaural systems over other techniques is that only two playback channels are required, and a *virtual* sound source can be positioned at any desired point in space with great accuracy. This advantage does not come without difficulties. The binaural signal has to be delivered to the listener's ears with great fidelity to be realistic. Generally headphones are used to render binaural signals, but in some applications this is not practical. As discussed later, techniques using standard stereo loudspeaker setups also exist, although their effectiveness is confined to very small sweet spots.

Another problem in binaural audio is the idiosyncratic nature of the binaural signal. Sound is *imprinted* with an acoustic signature particular to each listener before it reaches the ear drums. Differences in head and body dimensions contribute to audible acoustic changes that the brain learns to associate with the direction of the source. Thus, if a listener is presented with the binaural signal recorded at the ear canals of another person, localization errors generally increase (especially in the median plane) and realism and immersion will be somewhat compromised.

While there are still many unsolved problems and difficult challenges, practical VSS systems capable of rendering convincing spatial sound are already a reality. In developed countries, an increasingly larger segment of the population experiences virtual spatial sound on a regular basis. Many companies offer very convincing virtual sound rendering products for game playing and music listening applications. Governments in these countries also continue funding research and developing advanced VSS systems for aerospace and military purposes.

The field of binaural audio technology is vast and it would be very ambitious to attempt to cover every aspect of it in this chapter. Our goal is to outline the basic concepts related to this technology, and provide references relevant to each topic that we discuss. In Section 2 we give a brief introduction to the basics of spatial hearing. Next, in Section 3 we discuss the acoustics of spatial sound. Finally we discuss the design of VSS systems. Before we begin our discussion it is convenient to establish the scope of the applications that we will be considering in this chapter.

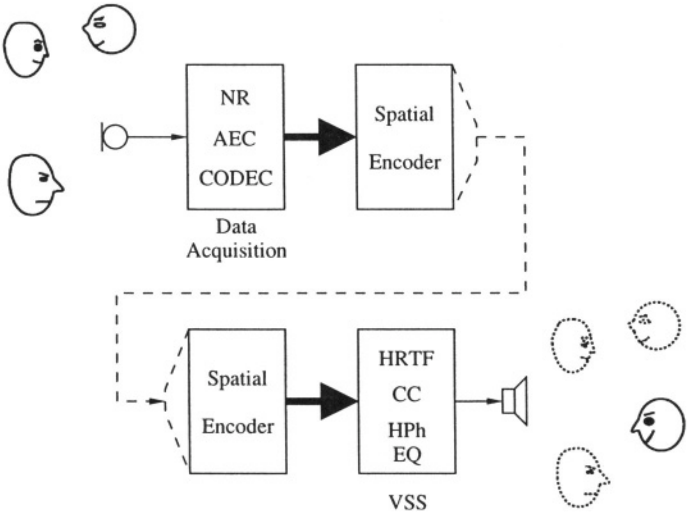


Figure 13.1 **Application of a VSS system in a tele-collaboration environment.**

1.1 SCOPE

The techniques that we describe constitute one particular way of rendering realistic soundstages in the tele-collaboration environment. Their applicability and effectiveness will depend on the particular application scenario and their implementation constraints. In a tele-collaboration environment, the goal of a VSS system is to provide the listener with a realistic immersion into a virtual environment shared with the other conference participants. Depending on the degree of realism with which this is achieved, the results will vary from increased speech intelligibility due to the virtual spatial separation of the various participants (i.e. reduction of the cocktail party effect [21]), to complete immersion. A general application scenario is depicted in Fig. 13.1. Some applications that correspond to this general scenario are for example:

- corporate meetings,
- business transactions,
- interactive multiplayer games,
- remote education,
- live interactive music recording/performance,
- virtual gatherings.

These applications have very different requirements with respect to soundstage rendering. In some instances (such as multiplayer games), headphone rendering is possible and appropriate, while in others (virtual gatherings) it is inconvenient or unnatural. Another design variable includes the noise level in the acoustic environment, which in some cases is low (e.g. corporate meeting) and in others it may be higher. The acoustics of the room (e.g. office, living room, studio, etc) will affect the experience in free-field listening conditions (e.g. loudspeaker rendering). Implementation issues such as latency and computational complexity are also limiting factors in some cases (e.g. live interactive performance, games, etc).

For the rest of the chapter we assume that some mechanism is in charge of encoding, transmitting and decoding the sound sources and information about their spatial distribution. The VSS system is either fed with the individual source signals and (optionally) information about their spatial distribution, or with a binaural signal where all sources are already encoded into their respective positions. In each instance, the design of the VSS system will face different challenges as we describe in Section 4.

2. SPATIAL HEARING

Humans and many other animal species have the remarkable ability of identifying (with various degrees of accuracy) the direction of a sound source originating from any point in three-dimensional space. It has been proposed that this ability evolved as a survival mechanism that allowed ancient organisms to aurally localize predators and/or prey. This mechanism relies on the availability of multiple sensory inputs, which in the case of humans consist of the two acoustic signals reaching the ears. The placement of the ears on the horizontal plane maximizes lateral differences for sound sources around the listener [12].

The human auditory system is very sophisticated and thus capable of analyzing and extracting most spatial information pertaining to a sound source from these two signals. However, the process of localizing a sound source is dynamic and often aided and complemented with other sensory inputs (e.g. visual and/or vestibular). As we will describe later on this chapter, the cross-modal nature of the sound localization process contributes to making synthesis of realistic virtual sound sources a particularly difficult engineering challenge.

We next describe some of the basic mechanisms of sound localization. Our goal is to familiarize the reader with the terminology and the basics of spatial hearing. For a more detailed and in-depth treatment of these topics the reader is referred to the literature provided.

2.1 INTERAURAL COORDINATE SYSTEM

We start our discussion by placing a hypothetical listener in the center of the sound field. We define an interaural-polar coordinate system whose origin is at the center of the head (see Fig. 13.2) and a rotation axis defined by the positions

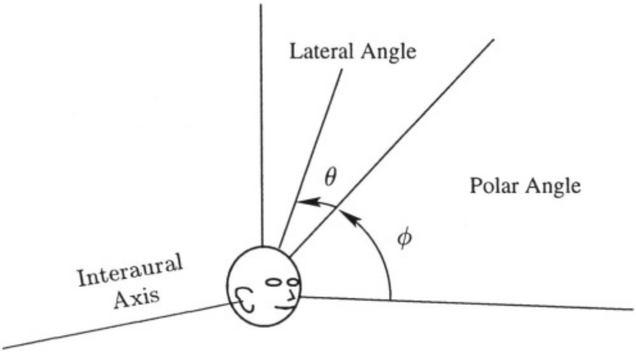


Figure 13.2 Interaural coordinate system.

of the two ear canals [23]. Here, the lateral angle θ is measured between a ray to the source and the median plane, and the polar angle ϕ is measured between the projection of that ray onto the median plane and the horizontal plane. The ear closest to the source is called *ipsilateral* and the opposite ear is called *contralateral.* In this system, $-90^\circ \leq \theta \leq 90^\circ$, where $\theta = -90^\circ$ corresponds to the left side of the listener, and $\theta = 90^\circ$ to the right side. The range of polar angle is $-90^\circ \leq \phi < 270^\circ$, where $\phi = -90^\circ$ is below the listener, $\phi = 0^\circ$ is in front, $\phi = 90^\circ$ is directly above of the listener's head, and $\phi = 180^\circ$ is in back.

Notice that different coordinate systems are used in the various studies found in the literature. One common system is the vertical polar system where the axis of rotation is defined in the up/down direction with respect to the subject. The choice of the interaural system for this discussion will become clear when we describe acoustic measurements in Section 4.2.

2.2 INTERAURAL DIFFERENCES

One of the basic binaural processing mechanisms involves the comparison between the time of arrival of the sound to the left and right ears. This process takes place in part of the auditory pathway from the cochlea to the auditory cortex known as the superior olive and can be modeled by cross-correlation between channels, followed by detection ([13], p. 144). The lag corresponding to the maximum peak of the cross-correlation function determines the perceived lateral angle of the sound source. This difference is commonly known as interaural time difference (ITD).

If we assume that the average distance between human ears is about 18 cm [22], the ITD has a maximum value of about ±0.75 ms. Notice that the ITD will not uniquely determine the direction of a sound source since there will exist ambiguity with respect to the front and back hemispheres (see Fig. 13.3). This

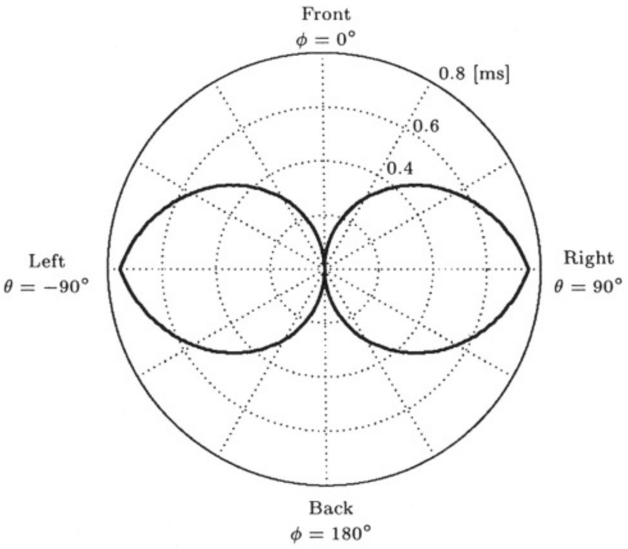


Figure 13.3 **Interaural time difference magnitude for a spherical head of radius $a = 9$ cm on the horizontal plane. The ITD function has opposite signs with respect to the median plane.**

ambiguity contributes to what is known as *front-back confusion* or front-back reversal.

The presence of the head in the sound field introduces a frequency dependency on the ITD. This dependency is accounted for by the subband analysis carried out in the auditory system. Several researchers have measured and modeled this frequency dependency [34, 17]. This dependency on frequency shows mainly at small lateral angles, where the ITD at lower frequencies (f< 1.5 kHz) is 50% larger than at higher frequencies. This discrepancy is not obvious and it is believed to be introduced by creeping waves at higher frequencies. See [35] for more details of the physics of these phenomena.

Another consequence of the presence of the head is that higher frequencies are attenuated or *shadowed* by the head as they reach the contralateral ear. This attenuation produces an interaural-level difference (ILD) which also plays a major role in lateral localization, especially at higher frequencies. Not only the contralateral signal suffers from low-pass filtering and attenuation, but the ipsilateral signal is boosted above a certain cut-off frequency (i.e. high shelf.) as the source moves laterally towards the interaural axis. If we assume that to a first degree of approximation the human head is spherical we can compute and plot the head shadow as a function of incidence angle as shown in Fig. 13.4 [24]. Notice the 6 dB boost for normal incidence. The ILD is computed by taking the ratio of the two shadow functions whose angles correspond to the

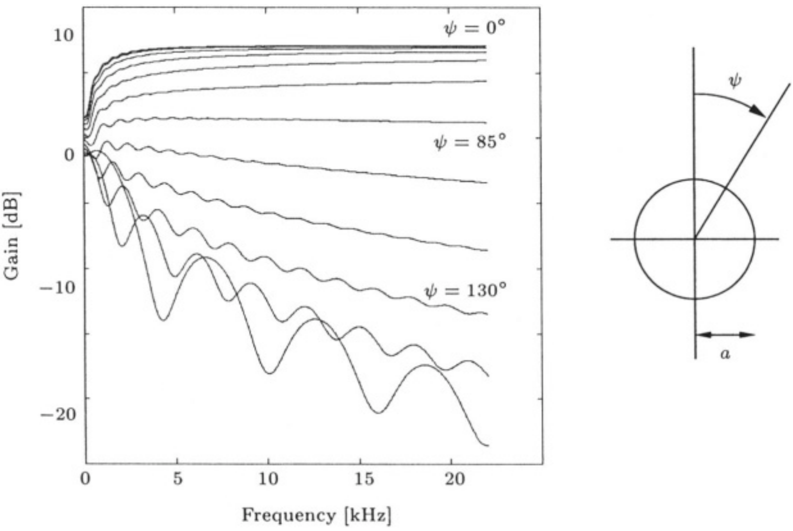


Figure 13.4 Magnitude response of a spherical head of radius $a = 9$ cm for various angles of incidence ψ. The response of only one ear at $\psi_0 = 0$ is shown.

angles of the source to each ear. Notice that the range of the head shadow will cause the ILD to have variations in the ±25 dB range.

The ITD and ILD are considered to be the primary cues for the perceived lateral angle of a sound source as proposed by Rayleigh in what is known as the Duplex theory [47]. For a given lateral angle, changes in the polar angle of a source cannot be predicted by the Duplex theory or obtained from a spherical-head model. For this reason the conical surface described described at a constant lateral angle for all possible polar angles is often called *cone of confusion*[1]. In principle, knowledge of the ITD and ILD would allow one to estimate the lateral angle, and hence to constrain the location of the source to a particular cone of confusion. Localization in elevation is well developed in humans, but involves other auditory cues as described next.

2.3 SPECTRAL CUES

In the median plane (i.e. $\theta = 0°$), the bilateral symmetry of the body implies that both the ITD and the ILD must vanish. However humans are still able to localize sound in the median plane by what is known as *monaural* cues, which are related to the spectral changes introduced by the outer ears (i.e. pinnae) at higher frequencies [48, 49, 7, 56] and other body structures like the torso at lower frequencies [8, 4]. With broadband sources an hypothesis for sound elevation localization is based on the notion that the listener is familiar with the timbre of the source, and decoding the polar angle involves comparing the memorized spectrum from the perceived spectrum. The spectral shape of the difference will determine the perceived polar angle. For narrowband sources, the perceived polar angle depends mainly on the frequency of the stimulus [13] (p. 45) and [18]. Effective localization of unfamiliar sources in the median plane can only be achieved with head motion.

The symmetry of the spherical model introduced in the previous section suggests that the monaural and ILD spectra should be independent of polar angle, which is contrary to fact in humans. Thus, it is likely that this dependence of spectra on polar angle can provide binaural as well as monaural cues to elevation. In the same way, the spherical-head model explains the *head shadow* part of the dependence of the ILD spectrum on lateral angle, but monaural and ILD spectra are rather complicated functions of lateral angle as well [23, 39]. If this is the case, then is it possible to use monaural cues to localize in lateral angle? Some studies have shown that monaural cues help monaural listeners (i.e. listeners with complete hearing loss in one ear) to localize the lateral direction of a source with relatively high accuracy. However, this was not the case for fully-binaural subjects with a blocked ear [39].

Spectral cues are also used to discriminate the front from the back when the sound source has sufficient high-frequency energy ($f > 3$ kHz). These cues are believed to be introduced by the front/back asymmetry of the pinna which result in a *pinna shadow* for sound sources arriving from the back. In the absence of this cue, head rotation is necessary to resolve front/back ambiguity [53].

2.4 DISTANCE CUES

Monaural spectra, ITD and ILD vary with lateral angle, polar angle, and distance. The range dependence is significant when the source is very close, but is relatively unimportant at greater distances. For distances of less than 1 m, the ILD increases significantly for lateral sources, but the ITD and spectral cues are similar to those of distant sources. With this evidence some authors have proposed that binaural cues play an important role for nearby sources [15, 16].

At large distances interaural differences and spectral cues are not reliable cues to estimate the distance of a source. One of the most useful cues for range estimation is loudness. It is well known that the loudness (and to a lesser degree, the spectra) of a sound source changes with distance . As with median-plane localization, the effectiveness of this cue depends on the familiarity of the listener with the source. For unfamiliar sound sources and distances larger than 3 m the perceived distance of a source is proportional to its loudness and not to the true distance [41, 12].

Other more salient cues for distance perception are the cues derived from the acoustic environment. Reverberation and/or reflections from nearby surfaces play a major role in distance perception. The ratio of reverberation (or reflected) to direct sound (D/R ratio) is a function of the relative distance between source and listener and the room acoustics. This cue can be more reliably used by listeners even if they have no familiarity with the particular sound source [38]. As we discuss later in Section4.3.1 in headphone listening conditions these cues play a major role for externalization.

2.5 DYNAMIC CUES

Sound localization is not restricted to static sources or listeners. Localization in everyday life includes head and body motion. As the head moves, the spatial cues will change according to the nature of the motion, and this will have a net effect on localization. Theories that study this phenomenon are often called *motional theories* [13]. Dynamic cues are extremely useful to resolve the ambiguities that static cues cannot handle. Many studies have shown that when subjects are allowed to move the head, localization blur and front/back reversals are significantly reduced [51]. Localization is frequently reinforced by other sensory inputs, such as visual and vestibular, which also have dynamic properties. Experiments have shown that listeners evaluate interaural differences at the same time as they move their head in relation to the direction of the source. All cues need to be consistent to produce the correct perception, and vestibular and visual cues carry information [53].

Unfortunately, most of the research in spatial hearing has focused on the static case and there are still many unanswered questions regarding the dynamic behavior of the spatial hearing system. In a VSS system, the adjustment or correction of the relative position between the listener and the virtual source as either moves, needs to be accurate and seamless to provide the same experience that the listener would have in real life. As we discuss later (Section 4.) this is achieved by monitoring the listener's motion and adjusting the system parameters accordingly.

3. ACOUSTICS OF SPATIAL SOUND

All acoustic cues used by the hearing mechanism for decoding spatial information are encoded in the binaural signal. Understanding the physical phenomena that originates the cues is essential in the design of modern VSS systems. In this section, we describe how spatial cues are acoustically encoded in the binaural signal. The effect of the body on the signal is specified by the head-related transfer function.

3.1 THE HRTF

In an anechoic environment, as sound propagates from the source to the listener, the different structures of the listener's own body will introduce changes to the sound before it reaches the ear drums. The effects of the listener's body are captured by the head-related transfer function (HRTF), the transfer function between the sound pressure that is present at the center of the listener's head when the listener is absent and the sound pressure developed at the listener's ear. The HRTF is a function of direction, distance, and frequency. The inverse Fourier transform of the HRTF is the head-related impulse response (HRIR), which is a function of direction, distance, and time.

In the time domain, the ITD is encoded in the HRIR as differences in the time of arrival of the sound between ipsilateral and contralateral side. This can be observed in Fig. 13.5 where we show the HRIR amplitude and HRTF magnitude as functions of polar angle for various lateral angles. Close to the median plane ($\theta = 10°$), the time of arrival of the wavefront is similar for both ears. However, as the lateral angle increases the time of arrival to the contralateral ear progressively exceeds that of the ipsilateral, thus increasing the ITD. The ILD is encoded as the level differences observed in the HRTF magnitude responses. Notice how the level difference is small near the median plane, and increases with lateral angle.

In the median plane, both ITD and ILD are very small, but there are strong spectral variations (i.e. monaural cues) that change with polar angle as shown in Fig. 13.6. Here we show the HRIR amplitude and HRTF magnitude for a human subject in the median plane. (The left and right ears are almost identical due to the symmetry of the subject thus only one side is shown.) In the time domain (i.e. HRIR) one can see the following features; the time of arrival of the wavefront creates the main pulse, which seems to arrive at the same time for all polar angles. Closer examination reveals that for directions above the subject, the main pulse is broadened due to diffraction around the top of the head. Following the main pulse, a pinnae *echo* or reflection appears as the second brightest feature in the HRIR. Notice that the time lag of the reflection changes with polar angle. The perceptual effect of this echo is apparent if we examine the HRTF magnitude, where it manifests as pinna *notches* (comb filtering). These notches are the most prominent elevation-dependent features in the HRTF at higher frequencies. In the midrange, the most prominent feature is a broad resonance, near 3 kHz and which has more energy in frontal directions. Wavelengths corresponding to this resonance are in the order of the size of the pinna, which acts as a parabolic energy collector in front and casts a shadow for sounds coming from back (i.e. pinnae shadow). This resonance can be observed in the time domain as ripples in the HRIR. While analysis of the HRIR reveals the acoustic origin of some of the spectral features, the importance of the detailed time structure seems to be minor as shown by some localization experiments where the phase spectrum of the HRTF was altered [33].

At lower frequencies, where wavelengths are comparable to the head and body size, the main elevation-dependent feature is a series of notches whose center frequency is lowest above the subject and highest towards the floor. This

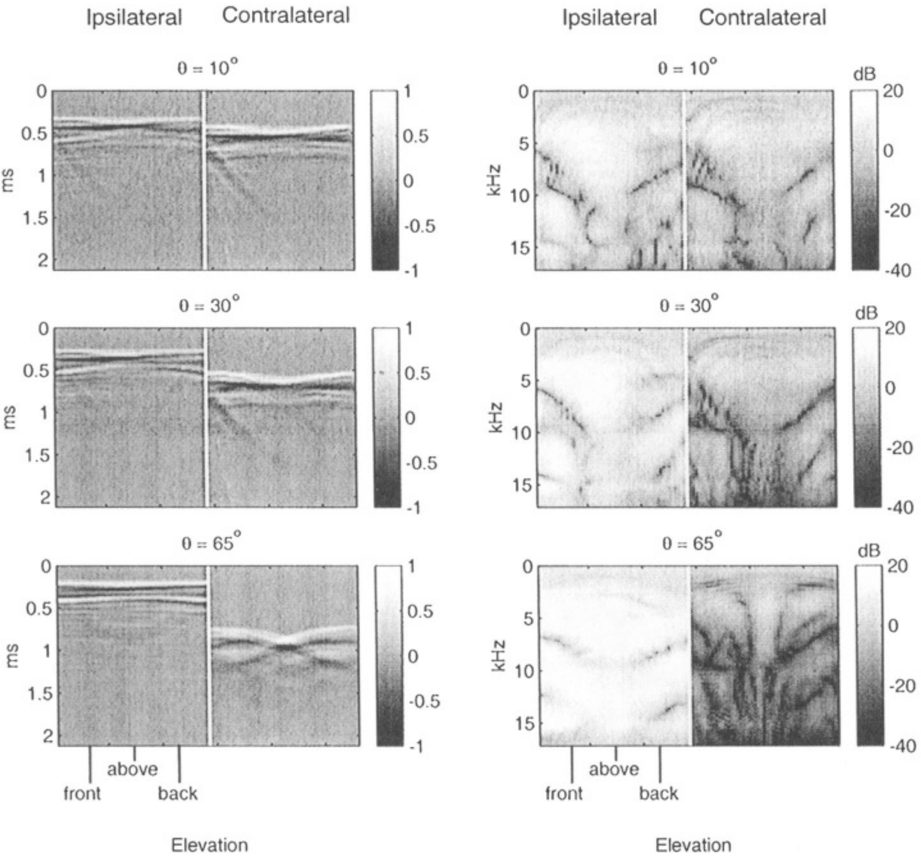


Figure 13.5 **The HRIR (left column) and HRTF magnitude (right column) for a human subject (subject 3 in the CIPIC database [3]). The abscissa in each panel is polar angle, the ordinate is time or frequency and the grayscale is amplitude or magnitude in dB.**

pattern is consistent with reflections arriving from the shoulders and torso, as can be seen in the HRIR as faint reflections at larger time lags. These features were until recently believed to have little importance. However, studies have suggested that cues derived from torso diffraction are effective for localizing low-frequency sources away from the median plane [8, 4].

For other lateral angles, the HRIR and HRTF show similar features. However, the differences between ipsilateral and contralateral sides increase, as can be seen in Fig. 13.5. While the HRTFs of most humans share these similarities, more detailed examination reveals subtle differences determined mainly by differences in body shape and size among subjects [45]. These subject-dependent differences have been shown to play a major role for precise localization. It is believed that only using ones own HRTF can result in realistic and accu-

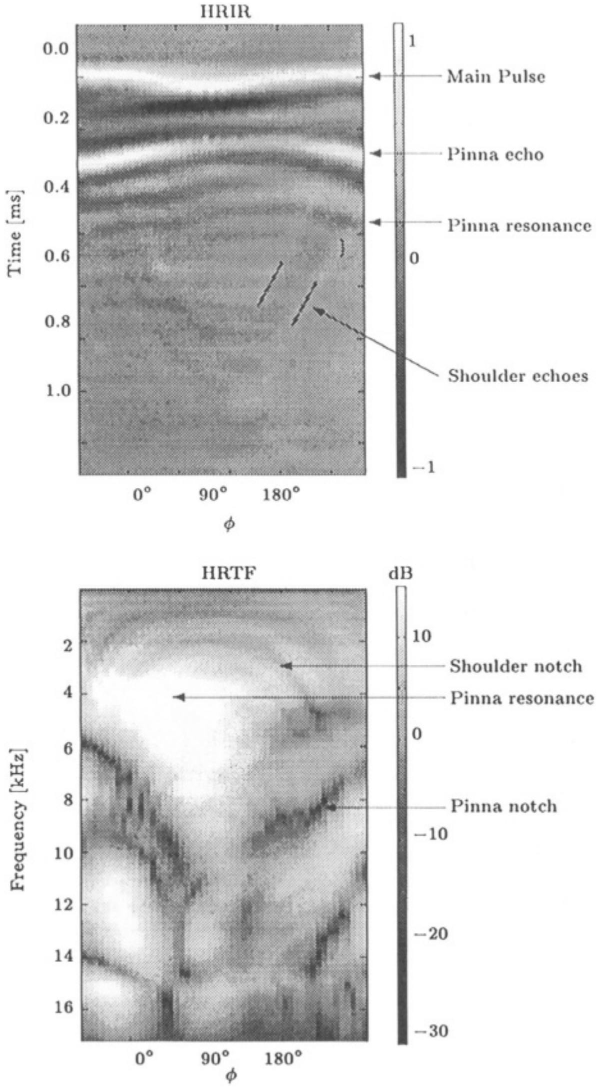


Figure 13.6 **The (a) HRIR and (b) HRTF magnitude in the median plane for a human subject. The abscissa is polar angle, the ordinate is time or frequency and the grayscale is amplitude in (a) and magnitude in dB in (b).**

rate binaural audio, as evidenced by various experiments [45]. Some studies have shown that some subjects can localize better with someone else's HRTF. However, these cases are rare exceptions [55].

3.2 ROOM ACOUSTICS

The acoustic modifications introduced by the HRTF are not the only components of the total spatial-hearing experience. Generally, listeners experience sound in rooms or acoustic environments that introduce additional distortions. These distortions can be either pleasing or annoying. For example, concert halls are carefully designed to create pleasing modifications for both musicians and audience. Everyday life environments are less carefully designed and can introduce undesirable modifications that in extreme cases can even impair speech communication.

Information about this acoustic environment can be drawn from analysis of its impulse response if one looks at it as a system. For a typical room, the impulse response shows the following three components:

1. *Direct path*: unmodified (or scaled) sound that reaches the ear via a direct linear path.

2. *Early reflections*: reflection from nearby surfaces such as the ceiling, the floor and the walls (within the first 100 ms).

3. *Late reverberation*: energy reaching the ear after multiple reflections from all surfaces.

The auditory system employs the acoustic information from the environment to determine, for example, the distance of the sound source (see Section 2.4). The direct path establishes the energy of the source, which is then used to estimate the D/R ratio from the total energy (see Section 2.4). The early reflections introduce timbre changes and echoes that give tonal quality to the room. The late reverberation increases the sense of spaciousness and is also used to estimate distance. Overall, the room acoustics contribute to the externalization and the realism of the sound source.

While a large fraction of the research on sound localization has been conducted in anechoic conditions, many studies have focused on sound localization inside rooms [29]. Under non-anechoic conditions there are psychoacoustic phenomena, such as the precedence effect (the perceived direction of a sound source corresponds to the direction of the first wavefront reaching the ears), that greatly affect sound localization. In general, the effect of the room is to increase the localization blur (i.e. uncertainty about the exact position of the source). Since correlated sound is arriving from several different directions, the auditory system receives conflicting cues that it somehow is able to resolve into a single direction estimate. However, this estimate has large variance compared to the estimate in anechoic conditions. Unless the listener is very close to a reflecting surface, localization in rooms is surprisingly accurate. Recent studies have shown that temporal integration and adaptation in higher auditory processes might be used to reduce blur and improve the estimate [50]. A VSS system needs to introduce or simulate room acoustic cues to achieve immersion and realism. This process, known as *auralization* will be described in the next section where we describe the design of a VSS system.

4. VIRTUAL SPATIAL SOUND SYSTEMS

While the concept of using the binaural signal to synthesize a realistic spatial experience is simple, the practical implications are not. In this section we discuss some issues in the design of practical VSS systems based on binaural signal processing.

4.1 HRTF MEASUREMENT

A key problem that arises in the design of a VSS system is the measurement of the HRTF. This is a long and expensive process, where human subjects (and sometimes mannequins) are equipped with microphones near or inside their ear canals to record the acoustic modifications introduced by the subject's anatomy. Generally the subject is in the center of some kind of apparatus (e.g. a rotating hoop), where fixed and/or movable loudspeakers placed in specified directions play the test signals. The recording is post-processed with knowledge about the test signal and the measurement system to compute the HRTF.

For such a measurement, it is quite desirable not to have the source location-dependent results be critically sensitive to the position of the point within (or outside) the ear canal where the HRTF is measured. Measurements near the eardrum will account for all individual localization characteristics of the listener including the ear canal resonance. However, such measurements are somewhat intrusive (and dangerous) and may not be necessary. It has been reported by many authors [43, 40, 37, 27], that all localization information can be obtained at a number of points within the ear canal (and possibly a few of millimeters outside). Binaural signals at the eardrum can be synthesized from HRTF measurements with the ear-canal blocked (so-called blocked-meatus measurements) along the ear canal if:

- The transfer function can be separated into its location-dependent and location-independent parts, where the location-dependent part can be measured with a blocked meatus.
- Blocked ear canal HRTF measurements with proper headphone compensation can reproduce the correct binaural signals at the ear drum.

In practice, acoustic measurement of the HRTF requires meticulous preparation and careful selection and/or design of the measurement equipment. It is not within the scope of this chapter to discuss these issues, but rather point the interested readers to some relevant literature [5]. Here we provide an example:

An HRTF measurement apparatus consists of the following components:

1. Sound generator.
2. Ear microphones (mounted at any position within the ear canal).
3. Loudspeakers (movable or many of them at desired measurement points).
4. Sound recorder (receives microphone pre-amplifier output)
5. Subject restrain or reference.
6. Signal processor.

The signal processor generates tests signals that are played back and recorded through the system. The recording is then processed to estimate the system response a system-identification technique. Among these techniques, Golay codes and Maximum Length Sequences (MSL) are widely used [5]. Although formally the HRIR has infinite duration, only a truncated response is estimated with these systems. Care is also taken to compensate for the linear distortions introduced by the loudspeakers and microphones. A reference measurement without the subject and with the microphones placed at the origin (free-field measurement) is inverted and applied as compensation. There are many factors that can reduce the fidelity of the HRTF measurement such as acoustic noise, electric noise, subject motion, apparatus inaccuracies, etc. All these factors need to be assessed and considered in the design of the system and the measurement protocols.

The measurement apparatus in our example is shown in Fig. 13.7. In this system several loudspeakers are mounted on a 1– m-radius hoop that rotates about the interaural axis (the trajectory described by each loudspeaker corresponds to one slice of the cone of confusion.) HRTF measurement is complicated and expensive. Fortunately, some institutions and universities around the world make HRTF measurements available to the public for research purposes, see for example [3].

An alternative or complement to acoustic measurements are numerical solutions where the HRTF is computed from analysis of the geometry of the subject's body. Photographs or three-dimensional laser scans are used to derive a geometry grid and numerical methods are applied to solve the wave equations at each point in the grid (e.g. boundary-element method (BEM) [31]. Proponents of these techniques argue that if made practical, this technology could replace acoustic measurements altogether. The advantage being that noise and subject motion are minimized. However, the acquisition of the detailed geometry of the human body and the computational complexity of these methods is still beyond practical limitations.

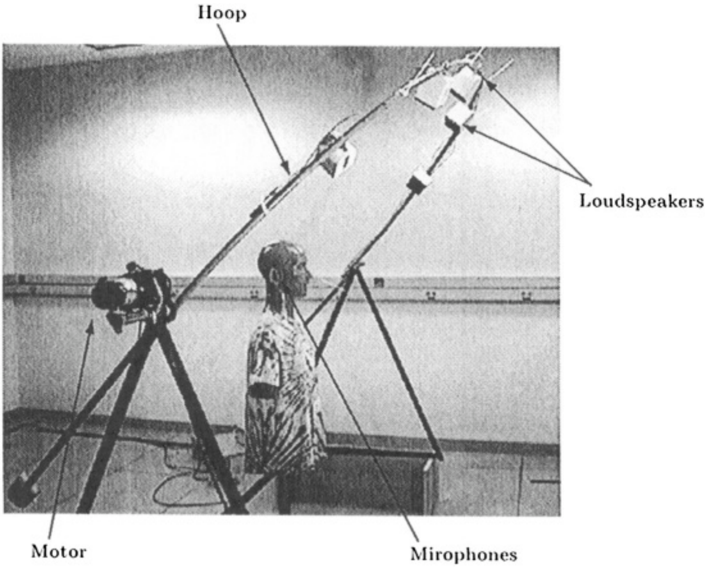


Figure 13.7 Example of an HRTF measurement system (courtesy of the CIPIC Interface Laboratory, UC Davis, California).

4.2 HRTF MODELLING

HRTF measurements are generally made available as a discrete set of finite impulse response (FIR) filter coefficients. Depending on the frequency resolution of the measurement, the length of the FIR filters will vary from a few hundred to a thousand samples (48 kHz sampling rate). The number of filters varies with the spatial resolution of the measurement apparatus, for example, to obtain a 5° spatial resolution in full 3-D space would require measurement of over 2,500 binaural responses.

A brute-force implementation approach for a VSS system is to use the measurements directly and perform real-time convolution (i.e. filtering) with the sound sources [2]. This approach is extremely expensive if one considers the duration of the HRIR and the high sampling rates required in high-quality audio. Another problem is related to the finite resolution of the measurement, which makes it necessary to interpolate filter coefficients to achieve smooth position changes.

As computational resources keep increasing, this approach would appear viable. However, rendering a large number of sources is also expected to be required as bandwidth and consumer expectations increase. Other more efficient ways of processing and rendering binaural signals are required for implementation of practical VSS systems. One common approach towards this goal is to model the HRTF or HRIR by a reduced number of parameters and to make the processing more efficient by operating in this parametric domain. There are two main approaches for modeling the HRTF, as discussed next.

4.2.1 Signal Models. In reality, the HRIR is an infinite impulse response system. However, as we discussed above in Section 4.2, due to practical limitations only a truncated impulse response can be measured. In some applications, the length of the measurement will prohibit its implementation by straight convolution. It is then advantageous to simplify these responses, either by reducing redundancy or by removing trivial information. Another advantage of modeling is that spatial interpolation is sometimes more conveniently done in the parameter space.

In essence, the signal models look at the HRTF or HRIR as a set of signals or filters and attempt to best represent the set with the least number of parameters (sometimes focusing on implementation constraints). Some examples of techniques are:

- Standard models where the minimum-phase part of the HRTF is used to derive filters and a delay line is used to model the frequency-independent ITD.
- Transform decompositions, such as principal component analysis (PCA) [20], spherical harmonics [19], etc. Interpolation of missing measurements works well with these models.
- Rational models of various kinds that attempt to model the data using system identification techniques. One example includes techniques where the HRIRs are modeled as IIR pole-zero filters with a set of common location-independent poles and location-dependent zeros [28].
- Perceptual models that remove perceptually-irrelevant information thus reducing the number of parameters required to represent the HRIR. An example are warped-frequency filter representations that take into account the non-uniform frequency resolution of the hearing system [30].

While these techniques achieve very good results, it is often the case that the models are not well-suited for customization to an individual listener. As discussed in Section 3.1, the HRTF depends strongly on the listener's anatomy, and a flexible system capable of adapting to different listeners by a simple change of parameters is an appealing tool. One modeling technique that is employed to do this is described next.

4.2.2 Structural Models. The main idea of a structural model is to factor the HRTF as a combination (parallel, serial or both) of independent parametric signal models. Each model represents a physical object that contributes to

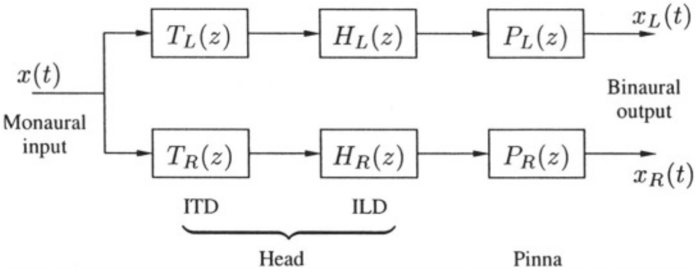


Figure 13.8 Signal flow of a typical structural HRTF model.

the acoustic modifications in the HRTF. As we have already seen, the sphere is a good model to describe the effect that the head has on the HRTF. The basic assumption in these models is an acoustic superposition principle where the different models do not overlap in space and/or frequency. For example, the sphere is a good model for the head whose major contribution is limited to the low-frequency part of the HRTF, but does not describe the behavior at high frequencies like a pinnae model would. Thus, a spherical-head model might be cascaded with a pinnae model to obtain an overall model valid for all frequencies.

A general signal flow diagram of a structural mode is shown in Fig. 13.8. Each submodel consists of simple signal processing operations, such as delays and or low-order IIR filter sections. Notice that one of the main advantages of these models is that the different submodels can receive data about the anthropometry of the listener and adjust its parameters accordingly. Many researchers have proposed structural models, for example:

- *Head models*: spherical head for ITD computation [57]; spherical head for ITD and head shadow computation [14]; spherical head with displaced ears for ITD computation [2]; ellipsoid with displaced ears for ITD computation [25].

- *Torso models*: spherical or ellipsoidal head-and-torso (HAT) [8] that uses ray-tracing arguments to model shoulder reflections; Snowman model (spherical head and torso) [1] that uses ray tracing arguments and models torso shadow as well.

- *Pinnae models*: sum of delayed energy to model pinna notches [46]; beamforming approaches [10, 20]; physical models of sound diffraction and reflection [36].

- *Other models*: contralateral pinnae models derived from ipsilateral HRTF [9].

4.3 VIRTUAL SPATIAL SOUND RENDERING

Once the HRTF or HRTF models have been selected and all sound sources have been filtered and combined into a binaural signal, the next step is to deliver this binaural sound to the listener. For the binaural technique to work, playback conditions need to match recording conditions. Two ways of achieving this are described next.

4.3.1 Headphone Rendering. The simplest way of delivering a binaural signal is through headphones. The reason is that each playback channel has to be delivered to the ears without contamination (i.e. crosstalk) to match the conditions under which the recording was made. However, the headphone itself will introduce linear distortions of its own and the sound reaching the ear drum will not necessarily be the sound that would have reached it under free-field listening. A compensation that accounts for the response and loading of the acoustic transducer and ear cup has to be applied in the case of headphones. This compensation or *headphone equalization* function will depend on many factors, including the measurement point and type of headphones, as well as the listener pinnae [44].

An acoustic circuit model such as that described in [42] can be used to compute the necessary headphone equalization functions for various measurement conditions. If the HRTF is measured at any position with an open ear canal (including close to the eardrum) the headphone-to-measurement transmission has to be compensated for. For blocked ear canal measurements, both the headphone-to-open and the headphone-to-blocked transmission at the position of measurement may be needed to devise the correct compensation. Some studies suggest that the scheme for binaural synthesis through headphones based on blocked meatus HRTF measurements and the sound transmission model in [42] is adequate to convey most localization cues available in free field hearing [5]. For HRTF's measured very near to the eardrum [55], headphone rendering necessitates one more compensation curve due to the ear canal resonance (which does not convey spatial information). Without compensation, the ear canal resonance will be excited twice, and will introduce unwanted timbre differences, that might even result in a loss of realism.

Headphone listening is a very unnatural experience. In an informal observation, a young two-year-old toddler who was experiencing music playback over headphones for the first time was extremely disturbed when turning her head did not result in the expected motion of the sound sources she was perceiving. After repeated head rotations, the toddler decided to leave the headphones alone and asked that her tape be played over the loudspeakers, arguing that she did not like how the sound *followed* her when she was wearing the headphones.

To achieve realism, binaural signal rendering through headphones requires that the spatial cues for each source be changed according to its relative position with respect to the listener. Otherwise, the source will appear to *follow* the listener when he or she rotates or moves the head. This compensation is possible only if some knowledge about the absolute position and head rotation angle of the subject are available. This is generally achieved by a head tracker, which is basically a transducer that monitors the position and rotation angles of the head with respect to a known reference. Head tracker technology includes electromagnetic sensors, gyroscopes, optical sensors, cameras, or even sound [52]. The requirements of the head tracker will depend on the application. In general, the tracker must have low latency (below 96 ms [54]) and be accurate to within a couple of degrees. Head trackers are in general expensive and inconvenient to wear, thus their use in consumer applications is very low.

For the VSS system to be able to compensate for head motion, the different sound sources and their respective spatial locations need to be available. If only a binaural signal is available, then the task becomes much more difficult since it would required source separation to recover the individual sources and their corresponding location.

Another requirement for realistic rendering is to add an acoustic environment to the binaural signal. This process, also known as *auralization* can be either a simple addition of reverberation (real or artificial) or a very complex room model. The main advantage of auralizing the binaural signal is that the sound sources are externalized. HRTF measurements are in general anechoic, and synthesis of a binaural signal will result in a dry and unnatural experience. While it is believed that realistic externalization is only possible with good room models and accurate motion compensation, some research has shown that under engineering constraints, a *canonical* environment can be designed to improve localization and reduce front-back reversals [6]. Auralization is a vast topic and cannot be covered with detail in this chapter, see [12] for more details.

4.3.2 Crosstalk-Cancellation Rendering. Another approach to rendering binaural signals that does not require headphones but instead uses a stereo loudspeaker setup is the so-called crosstalk canceller. These systems are based on a simple and elegant concept first introduced in the early 1960's by Shroeder and Atal [26] and Bauer [11]. The main idea is that in a stereo setup the acoustic *crosstalk* paths from left speaker to right ear ($h_{LR}(t)$) and right speaker to left ear $h_{RL}(t)$ (see Fig. 13.9) that naturally occur in free field conditions are measured and/or modeled, and used to compute a preprocessing filtering network. This network creates two signals per channel that upon reaching the ears cancel and reinforce each other in a way that only the desired binaural signals

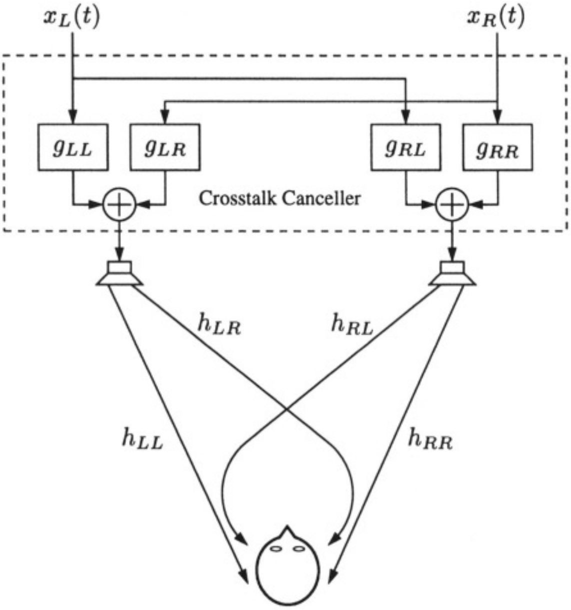


Figure 13.9 **Acoustic paths in crosstalk cancellation.**

at each ear prevail. Mathematically, the filters in the cancelling network $g_{ij}(t)$ are derived to satisfy the following set of equations:

$$h_{LL}(t) * g_{LL}(t) + h_{RL}(t) * g_{RL}(t) = \delta(t), \quad (13.1)$$
$$h_{LR}(t) * g_{LR}(t) + h_{RR}(t) * g_{RR}(t) = \delta(t), \quad (13.2)$$
$$h_{LL}(t) * g_{LR}(t) + h_{RL}(t) * g_{RR}(t) = 0, \quad (13.3)$$
$$h_{LR}(t) * g_{LL}(t) + h_{RR}(t) * g_{RL}(t) = 0, \quad (13.4)$$

where $*$ denotes convolution and $\delta(t)$ is a unit impulse. Generally, this system is solved in the frequency domain where convolution turns into multiplication.

The solution to this set of equations is not always guaranteed to exist or be unique. Moreover, the solution may also be unstable and/or non-causal. However, in recent years some techniques to deal with these issues have proven effective in the design VSS systems. While the crosstalk cancellation concept is simple, in practice it is not easy to achieve good results. Assuming that a correct and well-designed crosstalk network is available, the sweet spot is typically extremely small and movements of just a few millimeters from the center result in severe localization errors, specially at higher frequencies (e.g. $f > 3$ kHz).

It is well-known that the main limitation of this system is the size of the sweet spot. Some techniques that try to alleviate this exploit the geometry of the setup and place the loudspeakers very close to each other. This *stereo dipole* achieves wider sweet spots but at the cost of acoustic efficiency since the cross-talk network filters will have very high gains [32]. Some authors have proposed solutions based on monitoring the motion of the head (e.g. using a video camera) and adapt the TACC according to the relative position changes between the loudspeakers and the listener [26].

Another problem for TACCs is the listening environment. Reverberation, specially early reflections form surrounding structures, will add more acoustic paths than those depicted in the idealized scenario of Fig. 13.9. These additional paths will not be cancelled by the TACC and will degrade the binaural effect.

The choice of the HRTF used to model the direct and crosstalk paths has impact over the effectiveness of the system. Unfortunately, every listener will receive the best experience with its own HRTF, thus the system will have to be reconfigured each time it is used by a different listener. Given the difficulty and expense of measuring HRTFs, the structural modeling approach studied above (Section 4.2.2) seems attractive for real-world applications where simple anthropometric input by the user would automatically adjust a set of parametric HRTFs structures and allow correct computation of the network filters.

The TACC solution is then useful for a single-listener in a well-controlled environment. A major breakthrough in this area would considerably increase the usability of this technique in practical multi-listener VSS systems.

5. CONCLUSIONS

In this chapter, we have given an overview of several aspects relevant to the design of VSS systems for tele-collaboration applications. While the research and theory supporting this technology are well-established, practical issues continue to be the main challenge. Among them, we have seen that HRTF measurement and modelling have reached very advanced stages. In particular, we believe that structural models will play an important role in future systems given their low computational requirements and their flexibility to be customized to individual listeners. Rendering of binaural audio continues to be the main problem. As we discussed, head-tracking for headphone playback is necessary to recreate the experience that listeners have in real environments, but this technology is expensive. Current technology for free-field rendering is acceptable only in very few cases, and it is likely that wavefield synthesis techniques, although more expensive, will have more success for multi-user applications in the near future.

In spite of its drawbacks, the realism and immersive qualities of binaural technology will continue to make it the basis for state-of-the-art VSS in next-generation tele-collaboration systems.

Acknowledgments

The author would like to thank Ralph Algazi and Richard Duda for their valuable comments and suggestions, and Dennis Thompson at the CIPIC Interface Laboratory for providing some of the material presented in this chapter. This work would not have been possible without the sponsorship of Mark Dolson and Creative Advanced Technology Center.

Notes

1. Notice that in the interaural coordinate system for a constant lateral angle and range, the trajectory described by the source along the polar angle corresponds to a slice of the cone of confusion.

2. This approach is also known as Convolvotron due to the name of the first system of this type (developed for NASA by Crystal River Engineering).

References

[1] V. R. Algazi, R. O. Duda, and D. M. Thompson, “The use of head-and-torso models for improved spatial sound synthesis,” in *113th AES Convention,* paper 5712, Los Angeles, CA, Oct. 2002.

[2] V. R. Algazi, C. Avendano, and R. O. Duda, “Estimation of a spherical-head model from anthropometry,” *J. Aud. Eng. Soc.,* vol. 49, no.6, pp. 472-478 June 2001.

[3] V. R. Algazi, R. O. Duda, D. M. Thompson, and C. Avendano, “The CIPIC HRTF database,” in *Proc. IEEE WASPAA’01,* New Paltz, NY, 2001.

[4] V. R. Algazi, C. Avendano, and R. O. Duda, “Elevation localization and head-related transfer function analysis at low frequencies,” *J. Acoust. Soc. Am.,* vol. 109, pp. 1110-1122, Mar. 2001.

[5] V. R. Algazi, C. Avendano, and D. Thompson, “Dependence of subject and measurement position in binaural signal acquisition,” *J. Audio Eng. Soc.,* vol. 47, no. 11, pp. 937-947, Nov. 1999.

[6] E. J. Angel, V. R. Algazi, and R. O. Duda, “On the design of canonical sound localization environments,” in *113th AES Convention*, paper 5714, Los Angeles, CA (Oct. 2002).

[7] F. Asano, Y. Suzuki and T. Sone, “Role of spectral cues in median plane localization,” *J. Acoust. Soc. Am.*, vol. 88, pp 159-168, 1990.

[8] C. Avendano, R. O. Duda, and V. R. Algazi, “A head-and-torso model for low-frequency elevation effects,” in *Proc. IEEE WASPAA’99,* paper 9-4, New Paltz, NY, 1999.

[9] C. Avendano, R. O. Duda, and V. R. Algazi, “Modelling the contralateral hrtf,” in *Proc. AES 16th International Conference*, pp. 313-318, Rovaniemi, Finland, 1999.

[10] D. W. Batteau, “The role of the pinna in human localization,” *Proc. Royal Society,* London Ser.B 168, pp. 159-180.

[11] B. B. Bauer, “Stereophonic earphones and binaural loudspeakers,” *J. Acoust. Soc. Am.,* vol.9, pp. 148-151, 1961.

[12] D. R. Begault, *3-D Sound for Virtual Reality and Multimedia,* AP Professional, Ed., Chesnut Field, MA. 1994.

[13] J. Blauert, *Spatial Hearing,* Revised Edition, MIT Press, Cambridge, Massachusetts, 1997.

[14] C. P. Brown and R. O. Duda, "A structural model for binaural sound synthesis," *IEEE Trans. on Speech and Audio Processing,* vol. 6, pp. 476-488, Sept. 1998.

[15] D. S. Brungart, *Near-Field Auditory Localization*, Massachusetts Institute of Technology, doctoral dissertation, 1998.

[16] D. S. Brungart and W. M. Rabinowitz, "Auditory localization of nearby sources. Head-related transfer functions," *J. Acoust. Soc. Am.,* vol. 106, pp. 1465-1478, Sept. 1999.

[17] T. N. Buell and E. R. Hafter, "Discrimination of interaural differences of time in the envelopes of high-frequency signals: integration times,"*J. Acoust. Soc. Am.,* vol. 84, pp. 2063-2066, 1988.

[18] R. A. Butler, "The bandwidth effect on monaural and binaural localization," *Hearing Research*, vol. 21, pp. 67-73, 1986.

[19] S. Carlile and D. Pralong, "The location-dependent nature of perceptually salient features of the human head-related transfer functions," *J. Acoust. Soc. Am.,* vol. 95, pp. 3445-3459, June 1994.

[20] J. Cehn, B. D. van Veen and K. E. Hecox, "External ear transfer function modelling: a beamforming approach," *J. Acoust. Soc. Am.,* vol. 92, pp. 1933-1944, Oct. 1992.

[21] E. C. Cherry, "Some experiments on the recognition of speechwith one and with two ears," *J. Acoust. Soc. Am.,* vol. 25, pp. 975-979, 1953.

[22] Henry Dreyfuss Associates, *The Measure of Man and Woman*, Whitney Library of Design, New York 1993.

[23] R. O. Duda, "Elevation dependence of the interaural transfer function," in *Binaural and Spatial Hearing in Real and Virtual Environments*, edited by R. H. Gilkey and T. R. Anderson, pp. 49-75, Lawrence Erlbaum Associates, Mahwah, New Jersey, 1997.

[24] R. O. Duda and W. L. Martens, "Range dependence of the response of a spherical head model,"*J. Acoust. Soc. Am.* vol. 104, pp. 3048-3058, 1998.

[25] R. O. Duda, C. Avendano, and V. R. Algazi, "An adaptable ellipsoidal head model for the interaural time difference," in *Proc. IEEE ICASSP*, Phoenix AZ, pp. II-965-968, 1999.

[26] W. G. Gardner, *3-D Audio Using Loudspeakers,* Kluwer Academic Publishers, Boston, MA, 1998.

[27] D. Hammershøi and H. Møller, "Sound transmission to and within the human ear canal," *J. Acoust. Soc. Am.,* vol. 100, pp. 408-427, July 1996.

[28] Y. Haneda, S. Makino, Y. Kaneda, and N. Kitawaki, "Common-acoustical-pole and zero modeling of head-related transfer functions," *IEEE TSAP*, vol. 7, Mar. 1999.

[29] W. M. Hartmann, "Sound localization in rooms," *J. Acoust. Soc. Am.*, vol. 86, pp. 1366-1373, 1983.

[30] J. Huopaniemi and J. O. Smith III, "Spectral and time-domain preprocessing and the choice of modelling error criteria for binaural digital filters," in *Proc. AES 16th International Conference,* pp. 301-311, Rovaniemi, Finland 1999.

[31] Y. Kahana, P. A. Nelson, M. Petyt, and S. Choi, "Numerical model of the transfer function of a dummy head and the external ear," in *Proc. AES 16th International Conference,* pp. 330-345, Rovaniemi, Finland 1999.

[32] O. Kirkeby, P. A. Nelson, and H. Hamada, “Local sound field reproduction using two closely spaced loudspeakers,” *J. Acoust. Soc. Am.,* vol. 104, pp. 1873-1981, Oct. 1998.

[33] A. Kulkarni, S. K. Isabelle, and H.S. Colburn, “Sensitivity of human subjects of head-related transfer-function phase spectra,” *J. Acoust. Soc. Am.,* vol. 105, pp. 2821-2840, May 1999.

[34] G. F. Kuhn, “Acoustics and measurements pertaining to directional hearing,” in *Directional Hearing,* W. A. Yost and G. Gourevitch, Eds., pp. 3-25, Springer Verlag, New York, 1987.

[35] G. F. Kuhn, “Model for the interaural time difference in the azimuthal plane,” *J. Acoust. Soc. Am.,* vol. 62, pp. 157-167, July 1977.

[36] E. A. Lopez-Poveda and R. Meddis, “A physical model of sound diffraction and reflections in the human concha,” *J. Acoust. Soc. Am.,* vol. 100, pp. 3248-3259, Nov. 1996.

[37] S. Mehrgardt and V. Mellert, “Transmission characteristics of the external human ear,” *J. Acoust. Soc. Am.,* vol. 61, pp. 1567-1576, June 1977.

[38] D. H. Mershon and J. N. Bowers, “Absolute and relative distance cues for the auditory perception of egocentric distance,” *Perception*, vol. 8, pp. 311-322, 1979.

[39] J. C. Middlebrooks, “Spectral shape cues for sound localization,” *Binaural and Spatial Hearing in Real and Virtual Environments*, Gilkey R.H. and Anderson T.R., eds, pp. 77-97, 1994.

[40] J. C. Middlebrooks and J.C. Makous, “Directional sensitivity of sound-pressure levels in the human ear canal,” *J Acoust. Soc. Am.,* vol. 86, pp. 89-107, July 1989.

[41] J. A. Molino, “Psychophysical verification of predicted interaural differences in localizing distant sound sources,” *J. Acoust. Soc. Am.,* vol. 55, pp. 139-147, Jan. 1974.

[42] H. Møller, “Fundamentals of binaural technology,” *Applied Acoustics,* vol. 36, pp. 171-218, 1992.

[43] H. Møller, C. B. Jensen, D. Hammershoi, and M.F. Sorensen, “Design criteria for headphones,” *J. Audio Eng. Soc.,* vol. 43, pp. 218-232, Apr. 1995

[44] H. Møller, D. Hammershøi, C.B. Jensen, and M.F. Sorensen, “Transfer characteristics of headphones measured on human ears,” *J. Audio Eng. Soc.,* vol. 43, pp. 203-217, Apr. 1995.

[45] H. Møller, M. F. Sorensen, C. B. Jensen and D. Hammershøi, “Binaural technique: do we need individual recordings?” *J. Audio Eng. Soc.,* vol. 44, pp. 451-468, Nov. 1996.

[46] J. Pompei, “An efficient HRTF model based on interference from delayed energy,” in *105th AES Convention*, San Francisco, CA, Preprint 4806, 1998.

[47] J. W. S. Rayleigh, *The Theory of Sound,* Macmillan, London 1877. Second edition Dover Publications, NY, 1945.

[48] E. A. G. Shaw, “Acoustic response of external ear replica at various angles of incidence,” in *86th Meeting of the Acoustical Society of America,* pp. 1-19, 1973.

[49] E. A. G. Shaw, “Acoustical features of the human external ear,” in *Binaural and Spatial Hearing in Real and Virtual Environments*, edited by R. H. Gilkey and T. R. Anderson, pp. 25-47, Lawrence Erlbaum Associates, Mahwah, New Jersey, 1997.

[50] B. Shinn-Cunningham and K. Kawakyu, “Neural Representation of Source Direction in Reverberant Space.” in *Proc. IEEE WASPAA’03*, New Paltz, NY, 2003.

[51] W. R. Thurlow and P. S. Runge, “Effects of induced head movements on localization of direct sound,” *J. Acoust. Soc. Am.*, vol. 42, pp. 480-487, 1967.

[52] M. Tikander, A. Harma, and M. Karjalainen, "Binaural positioning system for wearable augmented reality," in *Proc. IEEE WASPAA'03,* New Paltz, NY, 2003.

[53] H. Wallach, "The role of head movements and vestibular and visual cues in sound localization," *J. Exp. Psycol.,* vol. 27, pp. 339-368.

[54] E. M. Wenzel, "Effect of increasing system latency on localization of virtual sounds," in *Proc. AES 16th International Conference,* pp. 42-50, Rovaniemi, Finland 1999.

[55] F. L. Wightman and D. J. Kistler, "Headphone simulation of free filed listening. I:stimulus synthesis," *J. Acoust. Soc. Am.,* vol. 85, pp.858-867, Feb. 1989.

[56] F. L. Wightman and D. L Kistler, "Factors effecting the relative salience of sound localization cues," in *Binaural and Spatial Hearing in Real and Virtual Environments*, edited by R. H. Gilkey and T. R. Anderson, pp. 1-23, Lawrence Erlbaum Associates, Mahwah, New Jersey, 1997.

[57] R. S. Woodworth and H. Schlosberg, *Experimental Psychology,* Holt, Rinehard and Winston, NY, pp. 348-361, 1962.

Index